

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки
Кафедра технічної кібернетики**

«На правах рукопису»
УДК 004.043

До захисту допущено:
Завідувач кафедри
_____ Ігор ПАРХОМЕЙ
«__» _____ 2020 р.

**Магістерська дисертація
на здобуття ступеня магістра
за освітньо-професійною програмою «Інформаційне забезпечення
робототехнічних систем»
зі спеціальності 126 «Інформаційні системи та технології»
на тему: «Розробка підсистеми персоніфікації користувача з
використанням голосової біометрії»**

Виконав:

студент II курсу, групи ІК-з91мп
Мазовіта Дмитро Олегович _____

Керівник:

Старший викладач
Польшакова Ольга Михайлівна _____

Консультант з нормоконтролю:

доцент, к.т.н., доц.,
Пасько Віктор Петрович _____

Рецензент:

старший викладач
Моргаль Олег Михайлович _____

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць інших
авторів без відповідних посилань.

Студент _____

Київ – 2020 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

Рівень вищої освіти – другий (магістерський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційне забезпечення робототехнічних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Ігор ПАРХОМЕЙ

«__» _____ 2020 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Мазовіті Дмитру Олеговичу

1. Тема дисертації «Розробка підсистеми персоніфікації користувача з використанням голосової біометрії», науковий керівник дисертації ст. викладач Польшакова О.М., затверджені наказом по університету від «26» жовтня 2020р. № 3133-с
2. Термін подання студентом дисертації 14.12.2020
3. Об'єкт дослідження – розпізнавання голосу людини комп'ютером.
4. Вихідні дані – персоніфікація користувача, алгоритми та методи голосового розпізнавання.
5. Перелік завдань, які потрібно розробити – Проаналізувати предметну область та поставити задачу. Проаналізувати існуючі системи голосової біометрії. Дослідити доступні класифікатори та методи знаходжень аномалій. Спроектувати схему та підсистему голосової біометрії. Розробити консольний додаток голосової біометрії. Провести маркетинг аналіз розробленого консольного додатку.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу – шість плакатів

7. Орієнтовний перелік публікацій – Мазовіта Д.О., Польшакова О.М. Сучасні методи біометричної автентифікації: огляд та визначення перспектив розвитку// XVI MATERIÁLY MEZINÁRODNÍ VĚDECKO - PRAKTICKÁ KONFERENCE «AKTUÁLNÍ VYMOŽENOSTI VĚDY», 22 - 30 червня, 2020

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Перевірка на співпадіння	доцент Лісовиченко О.І.		
Нормоконтроль	доцент Пасько В.П.		

9. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз предметної області	01.10.2020-04.10.2020	
2	Огляд та аналіз існуючих систем голосової біометрії	05.10.2020-09.10.2020	
3	Дослідження доступних класифікаторів та методів знаходження аномалій	10.10.2020-11.10.2020	
4	Проектування схеми консольного додатку	12.10.2020-18.10.2020	
5	Проектування підсистеми голосової біометрії	19.10.2020-25.10.2020	
6	Розробка консольного додатку голосової біометрії	25.10.2020-08.11.2020	
7	Розгляд консольного додатку як стартап-проекту та проведення для нього маркетинг-аналізу	09.11.2020-13.11.2020	
8	Оформлення пояснювальної записки	14.11.2020-30.11.2020	
9	Нормоконтроль	30.11.2020-06.12.2020	
10	Перевірка на співпадіння	01.12.2020-10.12.2020	
11	Другий передзахист	14.12.2020	
12	Захист	22.12.2020	

Студент

Науковий керівник

Мазовіта Д.О.

Польшакова О.М.

АНОТАЦІЯ

У представленій роботі розглянуто розробку підсистеми біометричного розпізнавання особи по голосу

Було проаналізовано предметну область та поставлена задача на проектування підсистеми. Проаналізовано існуючі системи голосової біометрії. Досліджено доступні класифікатори. Спроектовано схема та підсистема голосової біометрії. Розроблено консольний додаток голосової біометрії. Проведено маркетинг-аналіз розробленого консольного додатку.

Результатом виконаної роботи є розроблена підсистема, яка виконують поставлені перед нею задачі з високою точністю та продуктивністю.

Ключові слова: голосова біометрія, аутентифікація особи, економія обчислювальних ресурсів, швидка біометрія, mfcc, класифікатори, python.

Розмір пояснювальної записки – 85 сторінок, 25 ілюстрації, 24 таблиці, 4 додатки.

ABSTRACT

In this paper the development of a biometric facial recognition system by voice is considered

The subject area was analyzed and the task was set. Existing voice biometrics systems are analyzed. Available classifiers and methods for finding anomalies are investigated. The scheme and voice biometrics system were designed. A console application for voice biometrics has been developed. A marketing analysis of the developed console application was carried out.

The result of this work is a developed system that performs the tasks assigned to it with high accuracy and productivity.

Keywords: voice biometrics, face authentication, saving computing resources, fast biometrics, mfcc, classifiers, python.

The size of the explanatory note is 85 pages, 25 illustrations, 24 tables, 4 appendices.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до магістерської дисертації

на тему: Розробка підсистеми персоніфікації користувача з використанням
голосової біометрії

Київ – 2020 року

ЗМІСТ

ВСТУП	11
1. ОГЛЯД ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	12
1.1 Огляд предметної області.....	12
1.2 Огляд і аналіз систем біометричного розпізнавання по голосу	15
1.2.1 AMIS (Biolink).....	15
1.2.2 Voice Key («Центр речевих технологій»).....	16
1.2.3 VocalPassword (Nuance).....	20
1.3.Порівняльна характеристика систем голосового розпізнавання.	21
Висновки до розділу	22
2. ОГЛЯД МАТЕМАТИЧНИХ МЕТОДІВ ОТРИМАННЯ ТА РОЗПІЗНАВАННЯ ГОЛОСОВГО ВІДБИТКУ	23
2.1 Формування голосового відбитку методом MFCCs.....	23
2.2 Класифікатори для розпізнавання відбитку	25
2.2.1 K-nn (k-nearest neighbours classifier).....	26
2.2.2 MLP (Багатошаровий перцептрон)	27
2.2.3 SVM (метод Опорних векторів)	29
2.2.4 Древа ухвалення рішень	30
2.2.5 Random forest classifier (Випадковий ліс)	32
2.2.6 AdaBoost.....	33
2.2.7 Gaussian NB (Наївний Бассів класифікатор)	33

2.2.8 Прихована Марківська Модель (НММ)	35
2.3 Вибір класифікатора, порівняння.....	44
2.4 Методи виявлення статичних викидів	45
2.4.1 Local Outlier Factor (Локальний коефіцієнт аномалії).....	45
2.4.2. One-class SVM (Однокласовий SVM)	47
2.4.3.Isolation Forest.....	48
Висновки до розділу	49
3. ПРОЕКТУВАННЯ ПІДСИСТЕМИ ГОЛОСОВОЇ БІОМЕТРІЇ	50
3.1 Загальна структурна схема підсистеми голосової біометрії.	50
3.2 Схема модуля відповідального за навчання підсистеми голосом	51
3.3 Схема модуля відповідального за автентифікацію особи.	51
3.4 Опис роботи підсистеми.....	53
Висновки до розділу	54
4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	56
4.1 Мова програмування Python	56
4.2 Модулі та бібліотеки.....	57
4.2.1 SciPy и SciKit-learn.....	57
4.2.2 Matplotlib.....	61
4.2.3 NumPy.....	63
4.3 Опис роботи консольного додатку голосової біометрії.....	63
Висновки до розділу	67
5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ	68

5.1 Опис ідеї проекту	68
5.2 Технологічний аудит ідеї проекту	70
5.3 Аналіз ринкових можливостей запуску стартап-проекту	71
5.4 Розроблення ринкової стратегії проекту	78
5.5 Розроблення маркетингової програми стартап-проекту	80
Висновки до розділу	83
ВИСНОВКИ.....	84
ПЕРЕЛІК ПОСИЛАНЬ	85
ДОДАТКИ.....	87

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

МН – Машинне Навчання

ШНМ – Штучні Нейронні Мережі

K-NN – K-Nearest Neighbours

MFCCs – Mel Frequency Cepstral Coefficients

SVM – Support Vector Machine

ВСТУП

Мова – невід'ємний елемент людської діяльності, який дозволяє людині пізнавати навколишній світ, передавати свої знання і досвід іншим людям. Усне мовлення – це висловлювання в звуковій формі, які стають можливими завдяки голосовому апарату людини.

Будь-яка людина має свої особливі голосові характеристики, які визначаються індивідуальною будовою його голосових органів. В процесі спілкування людина здатна на підсвідомому рівні розрізняти голоси будь-яких людей, але тим не менш, це завдання для обчислювальної техніки є нетривіальним.

Відомо, що завдання розпізнавання особистості по голосу було поставлено більше 40 років тому, але досі тривають дослідження в даній області. За останні роки якість розпізнавання мовної інформації значно зросла, однак головна проблема автоматичного розпізнавання диктора в будь-якому середовищі досі далека від ідеального рішення. Тому актуальні не тільки дослідження вже існуючих алгоритмів, а й пошук нових рішень в даній області.

Завданням розпізнавання особистості по голосу є виділення, класифікація і реагування на мову людини з вхідного звукового потік. При цьому зазвичай виділяється дві підзадачі: ідентифікація і верифікація.

Розпізнавання особистості по голосу може знайти застосування в багатьох сферах: криміналістика та судова експертиза, безпека, банківські технології, електронна комерція.

Мова являє собою складний сигнал, який утворюється в результаті перетворень, що відбуваються на різних рівнях: семантичному, мовному, артикуляційному (рівень голосового апарату людини) і акустичному (рівень фізичних властивостей звуку). Відмінності в цих перетвореннях призводять до відмінностей у властивостях мовного сигналу. При вирішенні завдання розпізнавання диктора по голосу дані відмінності можуть використовуватися з метою виділення індивідуальних характеристик голосу людини.

1. ОГЛЯД ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд предметної області

Щороку з'являються все нові сервіси, які все глибше вбудовуються в наше суспільство. Часом вони зачіпають життєво важливі функції, несанкціонований доступ до яких може в значній мірі негативно відбитися на стані суспільства. Тому завдання забезпечення безпеки цих сервісів є найчастіше найважливішою при їх створенні. Для забезпечення безпеки систем використовуються різні прийоми, в числі яких знаходиться і однозначне визначення користувача і наданих йому прав. Найбільш часто при визначенні користувача використовується пара логін-пароль. Але коли потрібна підвищена безпека при вході в систему застосовують додаткові способи. Найчастіше одним з варіантів є розпізнавання особистості користувача по його біометричним даним, зокрема – голосу.

Завдання ідентифікації користувача по голосу була поставлена більш ніж 40 років тому, але дослідження в цій області триває і в даний час. Останнім часом ми спостерігаємо значне підвищення якості ідентифікації інформації, проте основна проблема автоматичної ідентифікації користувача в будь-якій сфері ще далека від ідеалу. Тому актуальні як дослідження вже існуючих алгоритмів, так і пошук нових рішень в цій області.

При вході в будь-яку систему з розмежованими правами доступу користувач проходить кілька етапів перевірки його спроб входу на безпеку і правомірність. Першим етапом є аутентифікація – це процедура перевірки автентичності користувача при спробі входу в систему. Далі починається ідентифікація – це процес призначення або пошуку відповідності індивідуального ідентифікатора (унікальної ознаки об'єкта) користувачеві. У разі успішного проходження аутентифікації користувач проходить процедуру авторизації, тобто. призначення

користувачеві прав на виконання певних дій в системі, а також процес перевірки прав при спробі виконання цих дій.

Для забезпечення безпечної і надійної процедури входу застосовуються додаткові заходи захисту: введення багатофакторної аутентифікації, застосування ідентифікаторів пов'язаних з реальним світом і за допомогою біометричних даних людини. Слід пояснити, що біометрія – це методи автоматичної ідентифікації та аутентифікації людини, на підставі його фізіологічних або поведінкових характеристик. Найбільш часто використовуються для аутентифікації відбитки пальців, фотографії райдужної оболонки ока та обличчя. Цей вибір обґрунтований тим, що дані способи біометричної аутентифікації найменш схильні до різних змін залежних від користувача і порівняно легко можуть бути отримані. Але те що інші способи менш затребувані в якості факторів аутентифікації не означає, що вони гірше передають індивідуальні особливості людини. Навпаки, такі як голос, хода і почерк набагато складніше підробити потенційному зловмиснику, так як вони є процесом, а не статичним зображенням, що отже є безумовною перевагою над іншими методами.

Розвиток у людини такого складного голосового апарату призвело до придбання їм голосу – здатності людини видавати різні звуки за допомогою звукових зв'язок, які при проходженні через них повітря створюють звукові коливання. Унікальні характеристики голосу купуються при проходженні звуку через природні резонатори: трахею, носові порожнини, ротову порожнину і гортань.

Завдяки голосу стало можливим поява і подальший розвиток мови, за допомогою якої людина отримала можливість будувати складну комунікацію з іншими людьми. Це відкрило нові можливості для передачі і зберігання інформації, що вкрай благотворно вплинуло на становлення людини як соціального виду. Мова використовується не тільки для обміну інформацією, але і для передачі емоційного

стану, а також для визначення особистості диктору. Останнє часто використовується людиною на інтуїтивному рівні і в поєднанні зі здатністю виділяти мову іншої людини серед сторонніх шумів для розпізнавання його мови дає дивовижні результати, які поки не досяжні для штучних систем. Але за останні 20 років в галузі розпізнавання мови людини спостерігається прогрес – спеціалізовані системи навчилися не тільки розпізнавати хто говорить, але і що говорить. Якщо програмне забезпечення для розпізнавання мови перейшло в нішу електронних помічників і персональних секретарів, то системи розпізнавання особи людини по голосу в сектор забезпечення безпеки доступу.

Ідентифікація користувача по голосу знаходить застосування в багатьох сферах: криміналістика і судова експертиза, антитерористичний моніторинг, безпека, застосування в системах розмежування доступу.

Розпізнавання особистості по голосу або розпізнавання диктора (англ. Speaker recognition) відрізняється від проблеми розпізнавання мови головним чином тим, що проводиться розпізнавання особистості диктора, а не його мови. Існує два типи систем: текстозалежні і текстонезалежні. Перші використовують фіксовані слова або речення, як на етапі верифікації, так і на етапі навчання. Другі не прив'язані до заздалегідь визначених висловлювань і працюють з висловлюваннями будь-якої довжини і змісту. Їх недоліки та переваги впливають з їх принципу роботи. Давайте розглянемо їх. Переваги текстозалежних систем розпізнавання диктора в тому, що такі системи порівняно легше створювати, тимчасові осі ідентифікованого диктора і записаного шаблону вирівняні, а також точність такого методу часто вище. Недоліками ж є неможливість застосовувати висловлювання відрізняються від уже заздалегідь закладених в систему і чутливість до природних змін в мові людини, які виникають через хвороби, зміни емоційного стану і так далі. Для текстонезалежних систем відчутною перевагою є відсутність прив'язки до певних фраз-паролів, тобто їх можна використовувати наприклад для

судово-медичних експертиз, де часто доступна один єдиний запис голосу диктора, якого необхідно впізнати. Інша перевага даного виду полягає в тому, що якщо заданий поріг розпізнавання не досягається, то користувачеві досить просто продовжити говорити довільне висловлювання, не повторюючи одну і ту ж фразу знову і знову. Слабким місцем таких систем є підвищена складність розробки.

1.2 Огляд і аналіз систем біометричного розпізнавання по голосу

1.2.1 AMIS (Biolink)

Інформаційна система мультибіометричної ідентифікації. Застосовується в масштабних проєктах державних і комерційних структур при обробці великих масивів даних про користувачів фінансових, транспортних, платіжних і т.д. сервісів, а також в державних системах паспортно-візових документів нового покоління. Підтримує ідентифікацію за відбитками пальців, зображеннями осіб, відеоряду, голосу. Надає можливість швидкого підключення будь-яких інших біометричних ідентифікаторів відповідно до вимог Міжнародної організації зі стандартизації (ISO) та Міжнародної організації цивільної авіації (ICAO).

Як масштабована система мультибіометричної ідентифікації BioLink AMIS реалізує наступні основні функції:

- реєстрація та обробка біометричних та інших персональних даних користувачів;
- пошук відомостей про користувачів за різними запитамі, на основі одного або декількох біометричних ідентифікаторів та іншої інформації (ПІБ, паспортні дані, структурований словесний портрет і т. д.);
- моніторинг і аудит, управління обробкою запитів, захист інформації циркулюючої в системі BioLink AMIS.

Модульній клієнт-серверній архітектурі BioLink AMIS притаманна висока гнучкість, що гарантує високу ефективність, продуктивність і масштабованість всієї системи мультибіометричної ідентифікації. Одночасно забезпечується можливість швидкого підключення додаткових модулів (наприклад, дозволяють розпізнавати користувачів за новим кодом), їх інтеграція не позначається на функціонуванні системи, не вимагає припинення її роботи або переконфігурації.

Основні компоненти BioLink AMIS:

- алгоритми біометричної ідентифікації за відбитками пальців, обличчям, голосом, почерком, а також алгоритм комплексної мультибіометричної ідентифікації, розроблений компанією BioLink;
- програмне забезпечення "проміжного" рівня (middleware) і сервіси біометричної ідентифікації, що реалізують згадані алгоритми;
- сервери бізнес-логіки та програмне забезпечення клієнтських робочих місць;
- кластери серверів біометричної ідентифікації та системи управління базами даних [3].

Загальна архітектура системи представлена на рис. 1.1.

Недоліки BioLink AMIS:

1. висока вартість;
2. в якості методу розпізнавання голосового відбитка використовуються нейронні мережі, що в свою чергу вимагає великих обчислювальних потужностей.

1.2.2 Voice Key («Центр речевих технологій»)

Технологія ідентифікації по голосу Voice Key заснована на унікальності геометрії мовного тракту кожної людини. У Voice Key використовується спектрально-формантний метод, що базується на різних спектральних характеристиках мови різних людей. Спектрально-формантний метод заснований

на аналізі поведінки трьох і більше формант, що відображають унікальність геометрії мовного тракту індивіда.



Рисунок 1.1. Загальна архітектура BioLink AMIS

Найбільш явно відмінність спектральних характеристик проявляється в співположенні формант в вокалізованих відрізках мови.

Використовуваний в Voice Key спектрально-формантний метод заснований на виділенні і порівнянні положення і динаміки поведінки трьох і більше формант. Використовується кілька десятків параметрів, що характеризують формантну структуру мови.

Принцип роботи включає 4 основних етапи:

1. створення шаблону – відомості про фізіологічну або поведінкову характеристику перетворюються в форму, доступну комп'ютерним технологіям, і зберігаються в пам'ять біометричної системи;
2. виділення – з пред'явленого ідентифікатора виділяються унікальні ознаки, аналізовані системою;
3. порівняння – зіставляються відомості про знову пред'явленому і раніше зареєстрованому ідентифікаторі;
4. рішення – вноситься висновок про те, збігаються або не збігаються знову пред'явлений і раніше зареєстрований ідентифікатор.

Технічна характеристика:

- мовно- та акценто-незалежна технологія аутентифікації;
- динамічно мінлива парольна фраза;
- аутентифікація по парольній фразі довжиною 3-5 секунд;
- адаптивна шумоочистка мовного сигналу;
- стійка робота в каналах зв'язку, що використовують стиснення мовного сигналу;
- (GSM, VoIP, ISDN);
- клієнт-серверна архітектура;
- механізми контролю якості та актуалізації еталонів;
- засоби адміністрування користувачів і аудиту аутентифікації.

Можливі варіанти застосування:

- аутентифікація дикторів в каналах телефонного зв'язку;
- голосові паролі для доступу до інтернет-ресурсів;
- підтвердження персоналом своїх дій при проведенні важливих операцій (доступ до даних, проведення транзакцій);

- контроль фізичного доступу в приміщення;

Переваги:

- немає необхідності пам'ятати довгі секретні символічні комбінації;
- не потрібні дорогі зчитувачі біометричної інформації;
- аутентифікація людини за індивідуальними біометричними ознаками, а не за виданим йому документом-картою, кодом, ключем, паролем і т. д. виключаються випадки несанкціонованих дій у разі втрати, крадіжки і передачі звичайного символічного пароля або електронного ключа;
- скорочуються витрати на обслуговування викликів і підвищення лояльності клієнтів;
- скорочення кількості «помилкових» з'єднань з оператором;
- можливість організації пріоритетних черг для VIP клієнтів;
- надання персоналізованої інформації в автоматичному режимі;
- природний і інтуїтивно зрозумілий для людини спосіб взаємодії з системою за допомогою голосу [4].

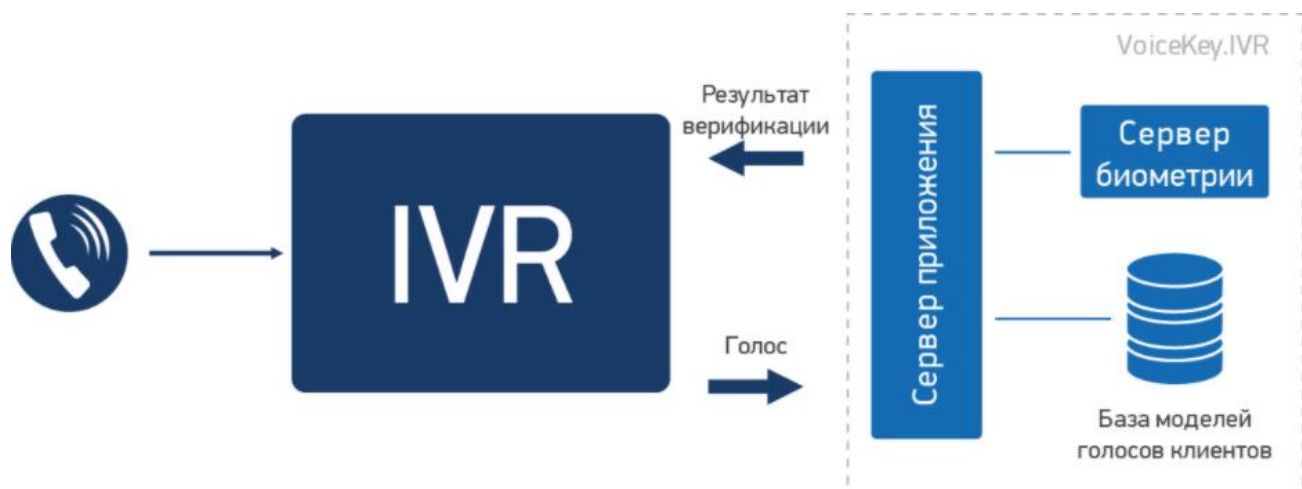


Рисунок 1.2. Загальна архітектура програми Voice Key

Недоліки Voice Key:

1. велика вартість;
2. володіє надлишком функцій, внаслідок чого має складне налаштування.

1.2.3 VocalPassword (Nuance)

Рішення для верифікації користувачів VocalPassword. Воно призначене для перевірки пароля або PIN-коду і підтвердження особи абонента по голосу. Система розпізнає промовляється код, по ньому отримує доступ до бази голосових відбитків і потім виявляє, чи відповідає інформація голосовому відбитку, що зберігається в базі. Цей продукт відноситься до класу систем ідентифікації із залежністю від тексту. Фактично це двофакторна ідентифікація по голосу і відомої користувачеві інформації. Щоб при цьому уникнути використання магнітофонного запису, пароль просять ввести кілька разів, і результати не повинні повністю збігатися. У підсумку можна досягти досить високих показників по надійності, проте клієнт повинен витратити на таку процедуру секунд 30. А якщо він до того ж забув пароль або назвав його неправильно, то процедура може розтягнутися [5].

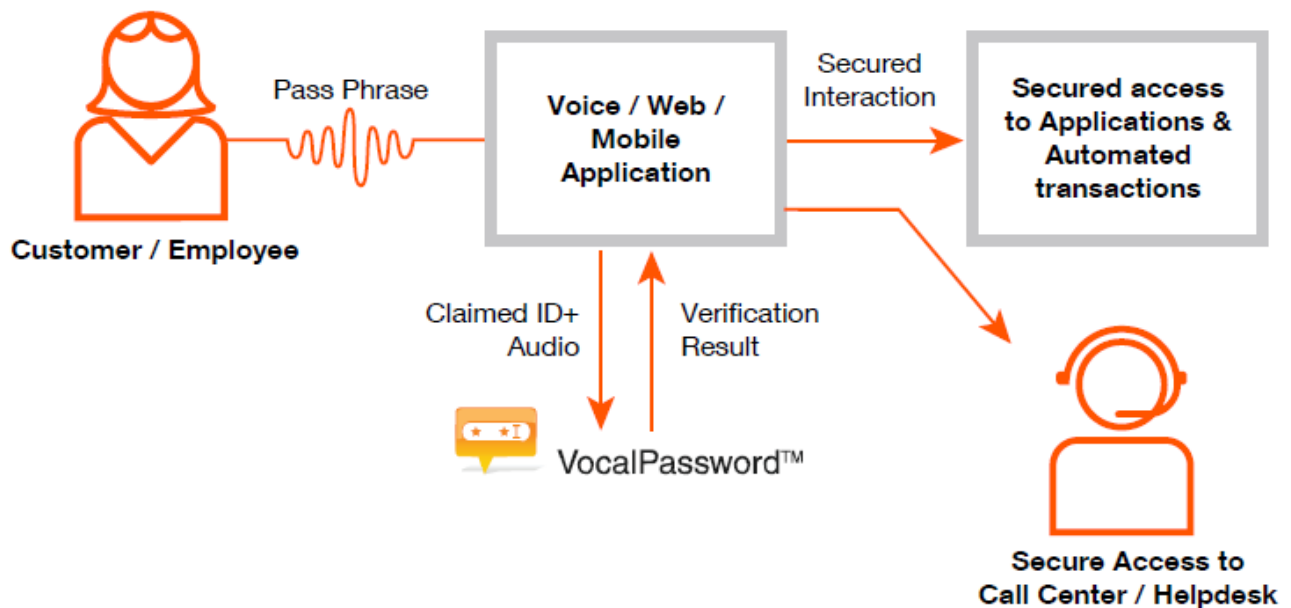


Рисунок 1.3. Загальна архітектура програми VocalPassword

Недоліки VocalPassword:

1. для проходження автентифікації клієнт має витратити 30 секунд, свого часу, що насправді є занадто довго.

1.3.Порівняльна характеристика систем голосового розпізнавання.

Розглянемо більш детально характеристики, інструменти розробки, використані методи, переваги та недоліки наведених вище систем.

Таблиця 1.1 Порівняльна таблиця систем голосової ідентифікації

Назва системи	BioLink AMIS	Voice Key	VocalPassword
Витяг ознак	MFCC	PLP, voicedness	MFCC PLP
Розпізнавання голосового відбитку	алгоритм bushderby	Алгоритм Витерби	Алгоритм Витерби
Точність	95%	98%	97%
Час розпізнавання голосового відбитку	5 сек.	12 сек.	30 сек.
Переваги	Великий відсоток точності розпізнавання.	Велика швидкість розпізнавання голосового відбитку.	Надійна система розпізнавання голосового відбитку
Недоліки	В якості розпізнавального алгоритму використовується нейронна мережа	Велика кількість функцій, тому важкий в налаштуванні.	Мала швидкість розпізнавання голосового відбитку

Висновки до розділу

В даному розділі був проведений огляд та аналіз предметної області по темі даної роботи, виконано моніторинг світового ринку біометричних технологій, який показав високий ступінь актуальності персоніфікації користувача в сучасних інформаційних системах, що вимагають високого рівня безпеки збереженої в них інформації.

В ході проведення порівняльної характеристики було виявлено такі недоліки:

- у багатьох існуючих комерційних і некомерційних систем голосової біометрії ймовірність помилки прямує до 1-5% за рахунок використання ШНМ, але більша частина методів та інструментів знаходиться під комерційною таємницею таких компаній як Nuance, Agnitio, VoiceVault та інших;
- використання цих систем для пересічного українця буде дещо дорогим. Так, цінова політика у провідних сервісах даної специфіки буде сягати сотень доларів США;
- в Україні занадто мало (або немає взагалі) комерційних, або некомерційних компаній, що займаються даною тематикою;

Виходячи з цього було поставлено задачу на магістерську дисертацію, а саме розробити підсистему, подібну до систем базованих на штучних нейронних мережах глибинного навчання у точності роботи і при цьому набагато менше вибагливу до обчислювальних ресурсів і більш швидко, здатну до використання у режимі реального часу на непотужних комп'ютерах, не витрачаючи кошти на оренду обчислювальних кластерів, або тим паче купівлю дорогих і об'ємних серверів.

Підсумовуючи можна сказати, що основною ціллю даної роботи є написання підсистеми, основними гарними якостями якою були б:

- невибагливість до обчислювальних ресурсів;
- швидкодія;
- і при цьому точність підсистеми, порівняна з точністю ШНМ.

2. ОГЛЯД МАТЕМАТИЧНИХ МЕТОДІВ ОТРИМАННЯ ТА РОЗПІЗНАВАННЯ ГОЛОСОВОГО ВІДБИТКУ

2.1 Формування голосового відбитку методом MFCCs

Останнім часом Глибокі Нейронні Мережі (ГНМ) користуються широким попитом. Незважаючи на це ГНМ не знайшла своє використання у даній магістерській дисертації, оскільки у цій роботі ми не використовуємо великих апаратних ресурсів. Проте, враховуючи, що головною метою цієї роботи – є створення швидкої підсистеми голосового розпізнавання, то доцільніше використати метод по отриманню голосового відбитку особи, який має високий рейтинг серед систем голосової біометрії, а саме MFCCs (Mel-Cepstral Coeficients). Опираючись на цей фактор, я вирішив використати метод MFCCs у своїй роботі.

Пояснимо метод та сферу його використання у даній роботі.

Для більш детального пояснення, метод MFCCs необхідно розділити на блоки, що продемонстровані на рис. 2.1.

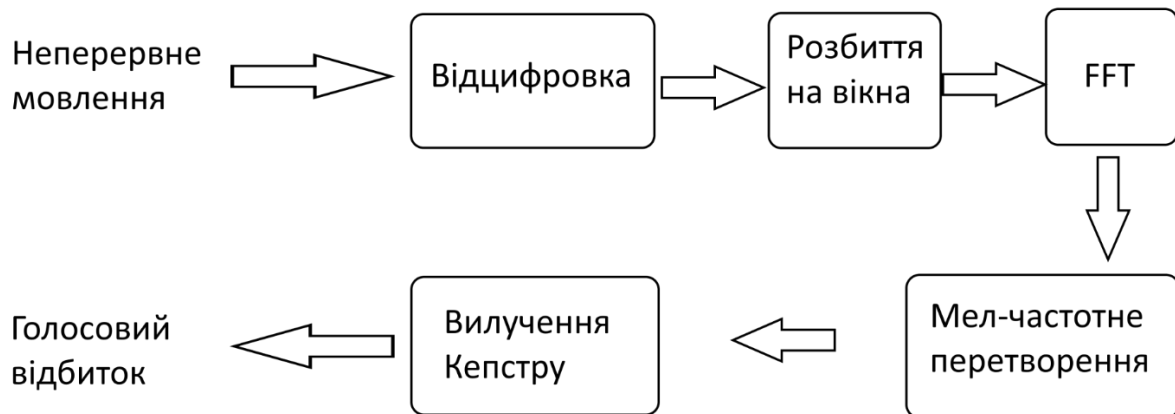


Рисунок 2.1. Схема методу MFCCs для вилучення голосового відбитку особи.

Вище ми зазначали, що метод MFCCs нам необхідний для вилучення голосового відбитку особи у період його безперервного мовлення. У подальшому

цей відбиток голосу буде представляти вектор векторів мел-кепстральних коефіцієнтів. Спочатку нам необхідно обрати якою буде кількість коефіцієнтів, адже нам потрібно, щоб вона була однаковою. Тобто у випадку, коли нам потрібно отримати відбитки голосу більшої кількості об'єктів, то нам необхідно мати ідентичну кількість коефіцієнтів у векторах відбитків. Виходячи з цього, ми можемо сказати, що одна із характеристик методу MFCCs – однакова кількість мел-кепстральних коефіцієнтів.

Таким чином, аналізуючи рис. 2.1, можна сказати, що першим етапом є прийом безперервного сигналу диктора, шляхом отримання звуку через мікрофон. Далі етап трансформації безперервного сигналу на дискретний (розбиття сигналу на дискрети, простіше кажучи на «семпли»), тобто етап відцифровки.

Потім ми отримуємо блоки семплів з певним набором, іншими словами «вікна». У подальшому, після фінального етапу роботи алгоритму, будь-яке вікно матиме вектор мел-кепстральних коефіцієнтів. За допомогою отриманого набору векторів є можливість навчити довільний класифікатор, проте про це трішки згодом.

Наступним етапом є отримання спектру, шляхом відокремлювання будь-яких частот у діючому сигналі. Fast Fourier Transform (FFT) є основою у такому перетворенні.

Зараз ми розглянемо основну гіпотезу методу MFCCs, яку ще називають мел-частотним перетворенням. Суть гіпотези полягає в тому, що частоти голосу особи, які ми отримали завдяки вищезазначеному перетворенню FFT, нам необхідно трансформувати у діапазон частот, що відповідає діапазону слуху людини. Іншими словами мел-діапазону, що має логарифмічне звукосприйняття людини.

На фінальному етапі ми використовуємо формулу 2.1., аби зі спектру сигналу диктора отримати спектр. Інакше кажучи, йде процес відокремлення частот із нашого частотного сигналу.

$$C_n = \sum_{k=1}^K (\log S_k) \left[n \left(k - \frac{1}{2} \right) \frac{\pi}{K} \right], \quad (2.1)$$

тут: C_n – кепстральний коефіцієнт n -го номера, S_k – складова вектору отриманого після мел-частотного перетворення і має номер k , K – кількість мел-кепстральних коефіцієнтів – параметр MFCCs, що був згаданий раніше, $n \in [1, K]$.

Отже ми матимемо відбиток голосу нашого об'єкта, що є вектором векторів дійсних чисел, про які ми говорили. Або ж, перефразуючи, ми отримали вектор мел-кепстральних коефіцієнтів індивідуально для кожного вікна. Тому, отриманий набір може слугувати як плато для навчання для будь-якого класифікатора МН.

2.2 Класифікатори для розпізнавання відбитку

В магістерській дисертації ми спробували використати майже всі класифікатори, що є в бібліотеці `scikit-learn` [10] мови програмування `python` (розглянемо її більш детально у розділі 4.2.1).

Перелік цих класифікаторів наведений нижче:

Перелік цих класифікаторів наведений нижче:

- K-NN (K-Nearest neighbours classifier, тобто K-найближчих сусідів);
- MLP (Multilayer perceptron, тобто Багатошаровий перцептрон);
- SVM (Support vector machine, метод Опорних векторів);
- Decision tree classifier (класифікатор на основі Дерев ухвалення рішень);
- Random forest classifier (класифікатор на основі Випадкового лісу);
- AdaBoost classifier (класифікатор на основі бустінгу під назвою AdaBoost);
- Gaussian NB (Gaussian Naive Bayes classifier, тобто Наївний баєсів класифікатор);

- НММ (Hidden Markov Model, тобто Прихована Марківська Модель)

2.2.1 K-nn (k-nearest neighbours classifier)

Метод k-найближчих сусідів використовується для вирішення завдання класифікації. Він відносить об'єкти до класу, якому належить більшість з k його найближчих сусідів в багатовимірному просторі ознак. Це один з найпростіших алгоритмів навчання класифікаційних моделей.

Число k – це кількість сусідніх об'єктів у просторі ознак, які порівнюються з об'єктом, що класифікується. Іншими словами, якщо $k = 10$, то кожен об'єкт порівнюється з 10-ма сусідами. Метод широко застосовується в технологіях Data Mining.

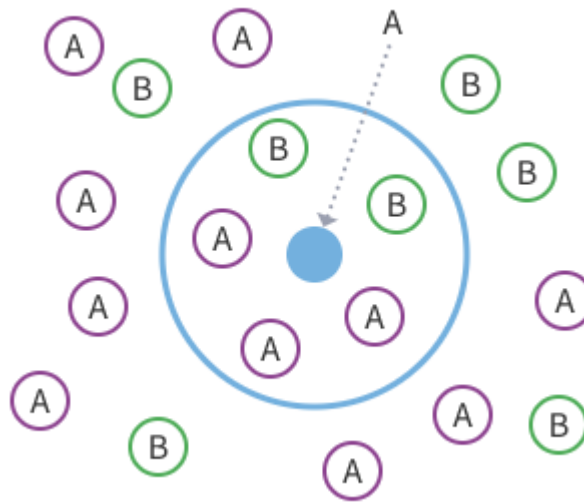


Рисунок 2.2. Візуалізація методу K-найближчих сусідів

В процесі навчання алгоритм просто запам'ятовує всі вектори ознак і відповідні їм мітки класів. При роботі з реальними даними, тобто спостереженнями, мітки класу яких невідомі, обчислюється відстань між вектором нового спостереження і раніше запам'ятованими. Потім вибирається k найближчих до нього векторів, і новий об'єкт відноситься до класу, якому належить більшість з них.

Вибір параметра k суперечливий. З одного боку, збільшення його значення підвищує достовірність класифікації, але при цьому межі між класами стають менш чіткими. На практиці хороші результати дають евристичні методи вибору параметра k , наприклад, перехресна перевірка.

Незважаючи на свою відносну алгоритмічну простоту, метод показує хороші результати. Головним його недоліком є висока обчислювальна трудомісткість, яка збільшується квадратично з ростом числа навчальних прикладів [11].

2.2.2 MLP (Багатошаровий перцептрон)

Багатошаровий перцептрон – це клас штучних нейронних мереж прямого поширення, що складаються як мінімум з трьох шарів: вхідного, прихованого і вихідного. За винятком вхідних, всі нейрони використовують нелінійну функцію активації.

При навчанні MLP використовується навчання з учителем і алгоритм зворотного поширення помилки.

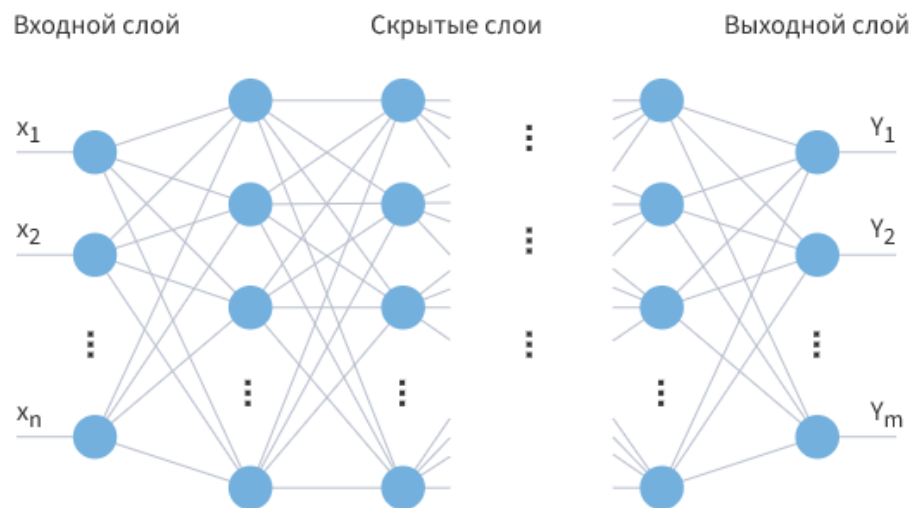


Рисунок 2.3. Візуалізація методу MLP

В якості активаційних функцій нейронів використовуються сигмоїдальні: логістичний або гіперболічний тангенс.

MLP показали можливість знаходити наближені рішення для надзвичайно складних завдань. Зокрема, вони є універсальним апроксиматором функцій, тому з успіхом використовуються в побудові регресійних моделей. Оскільки класифікацію можна розглядати як окремий випадок регресії, коли вихідна змінна категоріальна, на основі MLP можна будувати класифікатори.

Пік популярності MLP в машинному навчанні припав на 1980-і роки в таких областях, як розпізнавання мови і зображень, системах машинного перекладу. Однак пізніше вони зіткнулися з конкуренцією з іншими технологіями машинного навчання, такими, як машини опорних векторів. Інтерес до багатосарових перцептронів повернувся завдяки успіхам глибокого навчання.

Вперше багатосаровий перцептрон був запропонований Ф. Розенлаттом. Однак в тому вигляді, в якому він використовується в даний час, багатосаровий перцептрон був розроблений Д.Румельхартом.

Перцептрон Румельхарта відрізняється від перцептрона Розенблатта за такими властивостями:

- використання нелінійної активаційної функції;
- число прихованих шарів більше одного (зазвичай не більше трьох);
- вхідні сигнали не бінарні, а кодуються десятковими числами, нормованими до інтервалу $[0,1]$;
- вихідна помилка мережі визначається не як число помилково розпізнаних прикладів, а як деяке значення нев'язки;
- навчання проводиться не до мінімізації помилки, а до стабілізації ваг мережі, що дозволяє уникнути перенавчання.

В даний час багатосарові перцептрони як і раніше є популярними інструментом аналізу даних і входять в більшість платформ бізнес-аналітики. Але його використання потребує великих ресурсів [12].

2.2.3 SVM (метод Опорних векторів)

SVM (Support-Vector Machines – метод Опорних векторів) є методом Машинного Навчання з учителем, що використовується як для класифікації так і для регресії. На вхід даного алгоритму подається маркований вектор ознак, алгоритм навчається, після чого він готовий класифікувати (певними мітками) немарковані вектори ознак.

Найвідомішим різновидом SVM є бінарний SVM, який може навчитися розподіляти вектори ознак тільки на два класи. Основною задачею моделі бінарного SVM є представлення точок у просторі з максимальною відстанню між класами. Таким чином при навчанні модель знаходить межу поділу між двома класами. І вже на фазі розпізнавання метод Опорних Векторів може розподілити вектори ознак відповідно цієї межі на той чи інший клас.

Сама назва методу походить від назви граничних точок класів між якими проходить межа поділу, такі точки називаються опорними. А задача методу в максимізації відстані між цими опорними точками (векторами, якщо ми говоримо про багатомірний простір).

Графічно принцип роботи SVM зображено на рис. 2.4.

Існує поняття жорсткої та м'якої межі. У першому випадку межа представляє собою чітку лінійну пряму, у другому – така пряма є нечіткою. Таким чином якщо у другому випадку провести лінійну пряму, то вона б не розділила правильно вектори ознак на два класи.

До недоліків даного методу можна віднести:

- всі вхідні точки мають бути марковані;
 - напряму SVM можна використати тільки для бінарної класифікації;
- Хоча в цьому випадку можна застосувати алгоритми, що розподіляють багатокласову класифікацію на багато бінарних задач [13].

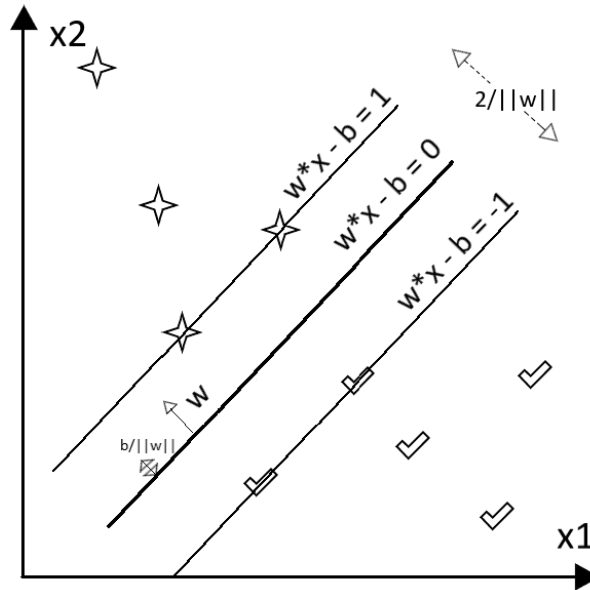


Рисунок 2.4. Принцип роботи SVM (Support-Vector Machines – методу Опорних векторів) на площині

Існує поняття жорсткої та м'якої межі. У першому випадку межа представляє собою чітку лінійну пряму, у другому – така пряма є нечіткою. Таким чином якщо у другому випадку провести лінійну пряму, то вона б не розділила правильно вектори ознак на два класи.

До недоліків даного методу можна віднести:

- всі вхідні точки мають бути марковані;
- напряму SVM можна використати тільки для бінарної класифікації;

Хоча в цьому випадку можна застосувати алгоритми, що розподіляють багатокласову класифікацію на багато бінарних задач [13].

2.2.4 Древа ухвалення рішень

В контексті даної роботи ми розглядаємо Древа ухвалення рішень як модель прогнозування, що застосовується у статистиці та МН. При навчанні моделі Древа рішень відбувається перехід від спостережень за предметом (представленим у

гілках дерева рішень) до висновків про цільове значення предмета (представлене на листях дерева).

Приклад навченої моделі Дерева ухвалення рішень можна побачити на рис. 2.5.

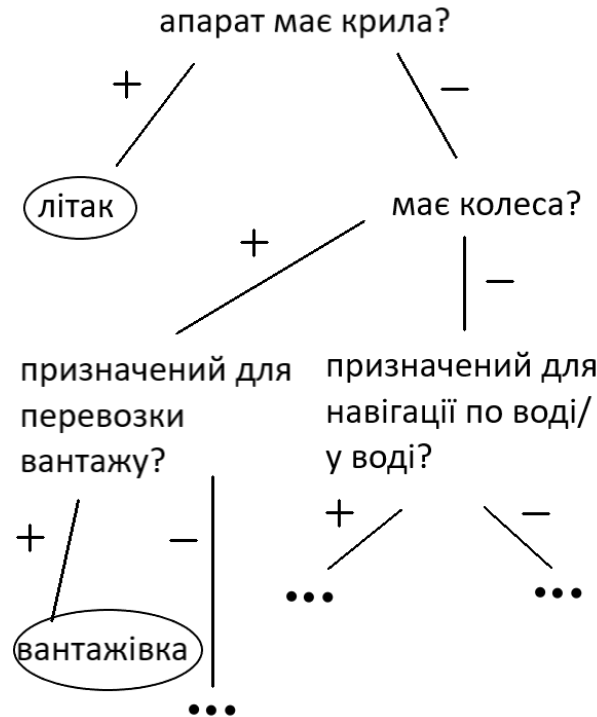


Рисунок 2.5. Приклад навченої моделі Дерева ухвалення рішень

Таким чином, дивлячись на рис. 2.5, можна збагнути, що при навчанні такої моделі відбувається побудова гілок дерева з кінцевими вузлами (листями). Ці вузли є ніщо іншим як певними класами, до яких відносяться певні вектори ознак. А вхідні вектори ознак це ніщо інше, як відповіді на поставлені питання у проміжних вузлах дерева.

Таким чином, знаючи структуру дерева та вхідний вектор ознак, можна чітко визначити листок (клас), до якого належить цей вектор.

Перевагами дерев ухвалення рішень є те що за допомогою такого підходу можна як класифікувати вхідні вектори ознак, так і побачити візуально покрокову структуру процесу прийняття рішень. Це корисно при аналізі даних, тому їх часто використовуються в цій області [14].

2.2.5 Random forest classifier (Випадковий ліс)

У машинному навчанні Випадковий ліс – це метод бустінгу, в основі якого закладено Дерева прийняття рішень [14] (метод МН, що був розглянутий у підрозділі 2.2.4). Його застосовують як для класифікації так і для регресії.

Перший алгоритм Випадкового лісу був запропонований Тінь Кам Хо у 1995 році.

Саме поняття “бустінгу” означає поєднання декількох достатньо простих моделей (методів) для створення однієї більш складної моделі. Так і метод, якому присвячено даний підрозділ дисертації утворений із декількох інших методів, а саме з декількох Дерев прийняття рішень. Дана “хитрість” дозволила збільшити точність, при цьому незначно втративши в наглядності дерев, вимогами до обчислювальних ресурсів та швидкості. Точність збільшується, оскільки зменшується варіативність моделі без збільшення її упередженості.

Набір моделей Дерев прийняття рішень навчається на різних частинах одного набору даних. Далі, при прогнозуванні, результати виходів кожної такої моделі агрегуються в одну, наприклад в таку, що зображена формулою 2.2.

$$f = \frac{1}{K} \sum_{k=1}^K f_k(x) \quad (2.2)$$

У формулі 2.2, K – це загальна к-ть простіших моделей (моделей Дерев прийняття рішень), f_k – функція класифікатора k -ї моделі, а x – це вхідний вектор ознак.

2.2.6 AdaBoost

Розшифровуючи AdaBoost отримаємо Adaptive Boosting (адаптивний бустінг) – це алгоритм МН, вперше розроблений Йоавом Фрейндом та Робертом Шапіром у 2003 році. Як і будь-який бустінг, AdaBoost використовує декілька “слабких” моделей задля побудови однієї більш “сильної”. Навідміну від Випадкового лісу, AdaBoost як слабкі моделі може використовувати не тільки Древа прийняття рішень (хоч дана конфігурація методу і вважається однією з найкращих для AdaBoost [15]).

Адаптивним даний метод називають тому, що кожен наступний простий класифікатор (модель) акцентує навчання на тих вхідних тренувальних векторах, з якими не впорався попередній. Окрім цього, цей метод в процесі навчання обирає лише ті ознаки, які підсилюють модель, викидаючи всі інші. Таким чином, даний алгоритм зменшує розмірність простору векторів і зменшує час роботи системи розпізнавання.

Однак, треба зазначити, що Адаптивний бустінг дещо чутливий до шуму.

2.2.7 Gaussian NB (Наївний Баєсів класифікатор)

Наївний Баєсів класифікатор є ймовірносним класифікатором, тобто в його випадку робиться припущення, що принцип появи вхідних векторів ознак є випадковим. Таким чином, якщо знати функцію розподілу, то є можливість кожному такому вектору віднести вірогідність приналежності до певного класу.

В основі даного класифікатора покладено теорему (формулу 2.3) Баєса.

$$P(\omega_k|x) = \frac{P(x|\omega_k)P(\omega_k)}{P(x)} \quad (2.3)$$

У формулі 2.3, $P(\omega_k|x)$ – це ймовірність того, що x належить класу ω_k ; $P(x|\omega_k)$ – це ймовірність того, що у класі ω_k існує образ x ; $P(\omega_k)$ – ймовірність класу ω_k ; $P(x)$ – ймовірність появи образу x .

Отже, як було вже сказано вище, щоб знайти ймовірність приналежності вектора ознак x до класу ω_k (тобто апостеріорну ймовірність), достатньо знати:

- функцію розподілу $P(x|\omega_k)$;
- відому (апріорну) ймовірність появи класу $P(\omega_k)$;
- відому ймовірність появи вектора ознак $P(x)$.

Часто ймовірність появи образу $P(x)$ відомо з постановки задачі. Щоб знайти апріорну ймовірність, зазвичай припускають, що всі класи рівноймовірні: $P(\omega_k) = 1/K$, де K – кількість наявних класів. А для функції розподілу просто роблять припущення щодо її виду. Найпопулярнішими є припущення у формі Багаточлену, у формі Бернуллі та у формі Гаусу [16].

В даній роботі було протестовано саме Гаусівську функцію розподілу, яку наведено формулою 2.4 нижче.

$$P(x|\omega_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}} \quad (2.4)$$

У формулі 2.4, маємо μ_k – середнє значення (математичне сподівання) образів x , що належать до класу ω_k ; σ_k^2 – дисперсія значень x , що належать до класу ω_k ;

2.2.8 Прихована Марківська Модель (НММ)

Hidden Markov Model (НММ) прихована марковська модель, НММ — це статистична марковська модель, у якій система, що моделюється, розглядається як марковський процес із неспостережуваними (прихованими) станами. НММ може бути представлено як найпростішу динамічну баєсову мережу. Математичний апарат для НММ було розроблено Леонардом Баумом зі співробітниками. Він тісно пов'язаний з більш ранньою працею про оптимальну нелінійну проблему фільтрування Руслана Стратоновича [23], який першим описав послідовно-зворотній алгоритм.

Марковський процес - це випадковий процес, конкретні значення якого для будь-якого заданого часового параметру $t+1$ залежать від значення у момент часу t , але не залежать від його значень у моменти часу $t-1$, $t-2$ і т. д. (дискретний випадок марковського процесу). Іншими словами «майбутній» стан j процесу залежить лише від «поточного» стану i , але не залежить від «минулого» (за умови, коли «поточний» стан процесу відомий).

Властивість, яка характеризує процес як марковський, називають марковською або властивістю Маркова. Вперше цю властивість було сформульовано російським математиком Марковим А. А., який 1907 року поклав початок вивченню послідовностей залежних випробувань і пов'язаних із ними сум випадкових величин. Цей напрямок досліджень відомий зараз під назвою теорії ланцюгів Маркова.

Таким чином, марківський процес може бути використаний для моделювання випадкової системи, яка змінює свій стан відповідно до правила переходу, що залежить від поточного стану. У простіших марковських моделях (таких як ланцюги Маркова) стан є безпосередньо видимим спостерігачеві, і тому ймовірності переходу станів є єдиними параметрами. У прихованій марковській

моделі стан не є видимим безпосередньо, але вихід, залежний від стану, видимим є. Кожен стан має ймовірнісний розподіл усіх можливих вихідних значень. Отже, послідовність символів, згенерована НММ, дає якусь інформацію про послідовність станів. Прикметник «прихований» насамперед стосується послідовності станів, якою проходить модель, а не параметрів моделі; модель все одно називають «прихованою» марковською моделлю, навіть якщо ці параметри відомі точно.

Традиційно, НММ визначається набором символів, що зображений формулою 2.5:

$$\lambda=(N,M,A,B,\pi), \quad (2.5)$$

де:

- N – число станів моделі;
- M – число відмінних символів спостереження одного стану, інакше розмір алфавіту моделі (the number of distinct observations symbols per state, i.e., the discrete alphabet size);
- $A = \{a_{ij}\}$ це $N \times N$ матриця розподілу ймовірності переходу з стану i в стан j (A is the $N \times N$ state transition probability distribution given in the form of a matrix $A = \{a_{ij}\}$);
- $B = \{b_j(k)\}$ це $N \times M$ матриця розподілу ймовірності спостереження k -го символу у стані j (the $N \times M$ observation symbol probability distribution given in the form of a matrix $B = \{b_j(k)\}$);
- π це вектор розподілу ймовірності початкового стану i (the initial state distribution vector $\pi = \{\pi_i\}$).

Простий приклад НММ наведено на рис. 2.6.

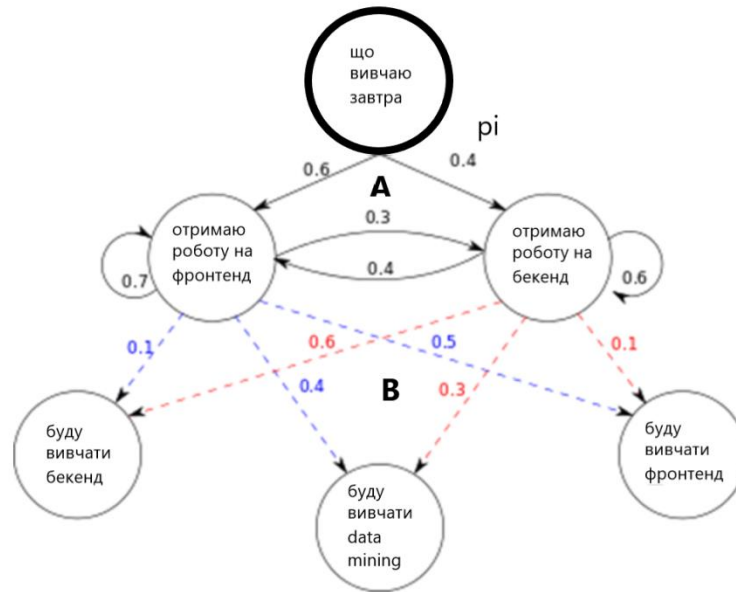


Рисунок 2.6. Приклад НММ: $\lambda=(2, 3, A, B, \pi)$, де 2- число станів моделі; 3 – алфавіт моделі; $A= \{a_{ij}\}$ – розподіл ймовірності переходу зі стану i в стан j ; $B = \{b_j(k)\}$ – розподіл ймовірності виходу k -го символу у стані j ; $\pi = \{\pi_i\}$ – вектор розподілу ймовірності початкового стану i .

Як правило, припущення Маркова дозволяє міркування і обчислення з моделлю, яка б в іншому випадку лишилась нерозв'язною.

Взагалі, існує чотири типи моделей Маркова, використовувані в різних ситуаціях, залежно від того, кожен послідовний стан спостерігається чи ні, і чи буде система скорегована на підставі зроблених спостережень, (табл.2.1)

Таблиця 2.1 — класифікація марковських моделей

	Стан системи, повністю спостережується	Стан системи частково спостережується
Система автономна	Ланцюг Маркова	Прихована марковська модель
Система управляється	Марковський процес ухвалення рішень	Частково спостережуваний марковський процес ухвалення рішень

Найпростіша модель Маркова це ланцюг Маркова. Він моделює стан системи з випадковою змінною, що змінюється в часі. У цьому контексті, припущення Маркова передбачає, що розподіл цієї змінної залежить тільки від розподілу в попередній стан. Приклад використання ланцюга Маркова - ланцюг Маркова Монте-Карло, який використовує властивість Маркова, щоб довести, що конкретний метод для виконання випадкового блукання буде зразком зі спільного розподілу системи.

НММ може використовуватися для вирішення задачі класифікації прихованих параметрів на основі спостережуваних. НММ являє собою кінцевий автомат, в якому переходи між станами здійснюються з певною ймовірністю, і задано стартовий стан, з якого починається процес. Через дискретні моменти часу може здійснюватися перехід в нові стани. При цьому кожному прихованому стану з заданою вірогідністю відповідає видимий стан. Крім того, поточний стан автомата залежить тільки від кінцевого числа попередніх, а закон зміни станів не змінюється в часі. Випадок, коли поточний стан залежить тільки від попереднього відповідає моделі першого порядку.

Визначаючи НММ через послідовності прихованих станів Q , та послідовність спостережень O основні параметри НММ можна подати у вигляді:

- множина прихованих станів $Q = \{q_0, \dots, q_N\}$, де q_0 - початковий стан, q_N - кінцевий стан;
- множина спостережень $O = \{o_1, \dots, o_M\}$;
- вхідний розподіл станів $\pi = \{\pi_i\}$, $1 \leq i \leq N$, який визначає ймовірність початку роботи в стані i ;
- матриця ймовірностей переходів між прихованими станами $A_{N \times N}$;
- $A(i, j) = a_{ij} = P(q_i, q_j)$, $1 \leq i, j \leq N$;
- матриця ймовірностей спостережень $B_{N \times M}$: $B(i, j) = b_{ij} = P(o_j | q_i)$,

$$1 \leq i \leq N, 1 \leq j \leq M.$$

Часову схему моделі тоді можна подати у вигляді як на рис. 2.7.

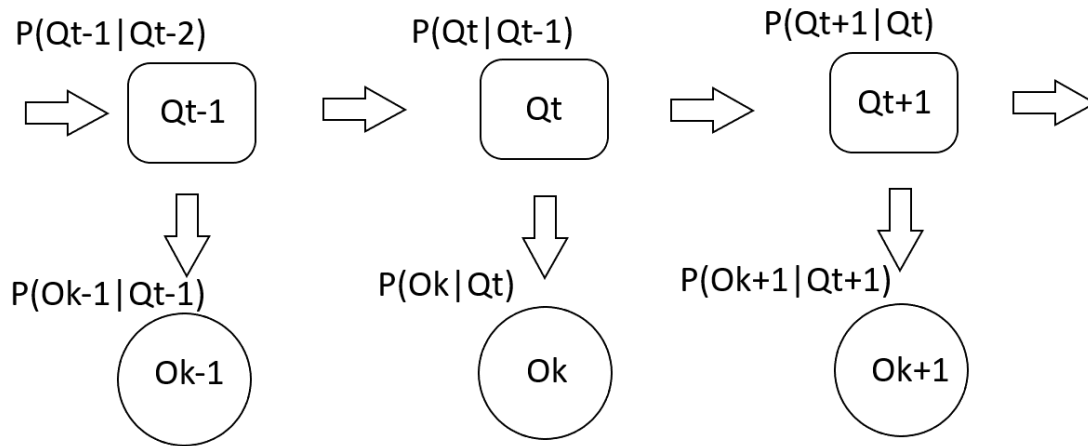


Рисунок 2.7. Загальна архітектура Hidden Markov Model

Як можна зрозуміти з рис. 2.7, прихована модель Маркова це ланцюг Маркова, для якого стан лише частково спостерігається. Іншими словами, спостереження, пов'язані зі станом системи, але їх, як правило, недостатньо, щоб точно визначити стан.

У стандартному типі прихованої марковської моделі, що тут розглядається, простір станів прихованих змінних є дискретним, тоді як самі спостереження можуть бути або дискретними (що зазвичай генеруються з категорійного розподілу) або неперервними (зазвичай з нормального розподілу). Параметри прихованої марковської моделі належать до двох типів, ймовірності переходів та ймовірності виходів. Ймовірності переходів керують тим, яким чином прихований стан у момент часу t обирається на підставі прихованого стану в момент часу $t-1$.

Вважається, що простір прихованих станів складається з одного з N можливих значень, змодельований як категорійний розподіл. Це означає, що для кожного з N

можливих станів, у якому прихована змінна може бути в момент часу t , є ймовірність переходу з цього стану до кожного з N можливих станів прихованої змінної в момент часу $t+1$, загалом N^2 ймовірностей переходів. Зауважимо, що набір ймовірностей переходів для переходів з будь-якого заданого стану мусить в сумі дорівнювати 1. Отже, матриця $N \times N$ ймовірностей переходів є марковською матрицею. Оскільки будь-яку одну ймовірність переходу може бути визначено, коли відомо решту, загальна кількість параметрів переходу складає $N(N-1)$.

До того ж, для кожного з N можливих станів є набір ймовірностей виходів, що керує розподілом спостережуваної змінної у певний момент часу для заданого стану прихованої змінної в цей момент часу. Розмір цього набору залежить від природи спостережуваної змінної. Наприклад, якщо спостережувана змінна є дискретною з M можливих значень, що регулюються категорійним розподілом, то буде $M-1$ окремих параметрів, загальним числом $N(M-1)$ параметрів виходу для всіх прихованих станів.

Виділяються три основні проблеми, які вирішуються за допомогою НММ:

1. (Evaluation) Обчислення ймовірності послідовності спостережень.

Проблема полягає у тому, щоб виходячі з заданих параметрів НММ, обчислити ймовірність певної вихідної послідовності. Це вимагає підсумовування над усіма можливими послідовностями станів, і це можна зробити ефективно за допомогою алгоритму Forward, який є формою динамічного програмування.

Для того, щоб визначити ймовірність появи заданої послідовності спостережень $O = \{o_1, \dots, o_R\}$, використовується наступний алгоритм (формули 2.6 – 2.8):

$$\alpha_0(i) = \alpha_{0i}, 1 \leq i \leq N, \quad (2.6)$$

$$\alpha_j(r) = \sum_{i=1}^N \alpha_i(r-1) a_{ij} b_{jt}, 1 \leq j \leq N, 1 \leq r \leq R, \quad (2.7)$$

$\alpha_j(r)$ – ймовірність того, що на r -му кроці модель виявиться в стані j . Тоді шукана ймовірність визначається за формулою:

$$P(O|A, B) = \sum_{i=1}^N \alpha_i(R) \quad (2.8)$$

2. (Decoding) Знаходження найбільш правдоподібної послідовності прихованих станів для спостережуваної послідовності.

Потрібно знайти найбільш правдоподібну послідовність прихованих станів $Q = \{q_1, \dots, q_R\}$ для заданої послідовності спостережень $O = \{o_1, \dots, o_R\}$, при якій досягається $\max P(O|Q)$. Це завдання вирішується за допомогою алгоритму, подібного алгоритму з попереднього пункту з тією лише різницею, що на кожному кроці запам'ятовується стан i , в якому $\alpha_i(r)$ приймає найбільше значення. В результаті вибирається послідовність станів, для якої $\alpha_i(R)$ приймає найбільше значення. Цей алгоритм називається алгоритмом Вітербо [16].

3. (Learning) Підбір параметрів моделі по заданій послідовності спостережень і довільній кількості прихованих станів.

Виходячи з заданої послідовності спостережень, чи послідовності послідовностей потрібно обчислити матриці A і B . Для вирішення даного завдання застосовується алгоритм Баума-Велша [18]. Якщо НММ задана набором $\lambda=(A, B, \pi)$, то алгоритм Баума-Велша знаходить таку λ^* , яка максимізує ймовірність вектору спостережень O (формула 2.9):

$$\lambda^* = \max_{\lambda} P(O | \lambda). \lambda \quad (2.9)$$

Розглянемо модель: $\lambda=(A, B, \pi)$, з випадковими початковими умовами.

Алгоритм ітеративно відновлює набір λ до сходження в одній точці

Пряма процедура (Forward):

Задамо формулу 2.10:

$$\alpha_i(t) = p(O_1 = o_1, \dots, O_t = o_t, Q_t = i | \lambda), \quad (2.10)$$

що є ймовірністю отримання заданої послідовності O_1, \dots, O_t для стану i в момент часу t . $\alpha_i(t)$ можливо обчислити рекурсивно (формула 2.11):

$$\begin{aligned} 1. \alpha_i(1) &= \pi_i \cdot b_i(O_1); \\ 2. \alpha_j(t+1) &= b_j(O_{t+1}) \sum_{i=1}^N \alpha_i(t) \cdot a_{ij} \end{aligned} \quad (2.11)$$

Зворотня процедура (Backward):

Ця процедура дозволяє обчислити (формула 2.12)

$$\beta_i(t) = p(O_{t+1} = o_{t+1}, \dots, O_T = o_T | Q_t = i, \lambda), \quad (2.12)$$

ймовірність кінцевої заданої послідовності O_{t+1}, \dots, O_T за умови, що ми почали із стану i в момент часу t . Далі за допомогою теореми Байеса обчислюються так звані

тимчасові змінні (відповідні ймовірності знаходження моделі у станах i та i, j які дають спостереження відповідно Q_t та Q_t, Q_{t+1}) (формула 2.13):

$$\begin{aligned}\gamma_i(t) &\equiv p(Q_t = i \mid O, \lambda) = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^N \alpha_j(t)\beta_j(t)}, \\ \xi_{ij}(t) &\equiv p(Q_t = i, Q_{t+1} = j \mid O, \lambda) = \frac{\alpha_i(t)a_{ij}\beta_j(t+1)b_j(o_{t+1})}{\sum_{i=1}^N \sum_{j=1}^N \alpha_i(t)a_{ij}\beta_j(t+1)b_j(o_{t+1})}\end{aligned}\tag{2.13}$$

За допомогою цих змінних можливо визначити значення матриць A , B , та вектора π (формула 2.14):

$$\begin{aligned}\bar{\pi}_i &= \gamma_i(1), \\ \bar{a}_{ij} &= \frac{\sum_{t=1}^{T-1} \xi_{ij}(t)}{\sum_{t=1}^{T-1} \gamma_i(t)}, \\ \bar{b}_i(k) &= \frac{\sum_{t=1}^T \delta_{O_t, o_k} \gamma_i(t)}{\sum_{t=1}^T \gamma_i(t)}.\end{aligned}\tag{2.14}$$

Використовуючи нові значення A , B і π ітерації продовжуються до сходження. Таким чином за певне число кроків буде досягнуто локальний оптимум, що дозволить вирішити поставлену задачу.

2.3 Вибір класифікатору, порівняння

Класифікатор, виконує функції ідентифікатора (знаходить найбільш схожу особу, із тих якими була навчена система перед цим, до особи яка проходить аутентифікацію) та першого верифікатора. Відповідно до ступення схожості, обирається дія системи до особи що проходить аутентифікацію (Дозволити доступ/ Заборонити доступ/ Запропонувати спробувати ще раз).

Класифікатор є одним з найважливіших параметрів фінального консольного додатку голосової біометрії. В рамках даної роботи була виконана оптимізація класифікатора, а саме було протестовано сім різних класифікаторів, алгоритми яких доступні в бібліотеці аналізу даних `scikit-learn`.

В якості набору даних для експерименту було взято 80 голосових зразків 40-ка різних дикторів.

Критерієм точності для порівняння класифікаторів були взяті значення `Allow-True` (кількість вірно розпізнаних голосових зразків осіб, тобто вірно допущених осіб; чим більше даний показник тим краща точність класифікатора) та `Allow-False` (кількість помилок першого роду при аутентифікації, тобто невірно допущених осіб; чим менше цей показник, тим краще точність класифікатора).

Наглядне порівняння точності класифікаторів зображено на рис. 2.8.

Відповідно до рис. 2.8, можна побачити, що найбільш точним в даному експерименті виявився класифікатор K-NN (k-nearest neighbor – к-найближчих сусідів). За друге місце змагаються класифікатори Naive Bayes та Random Forest. Найменш точним в даному експерименті виявився класифікатор SVM (Support-Vector Machine).

Тому саме класифікатор K-NN був вибраний як раціональний в даній роботі. Саме він і був використаний як ідентифікатор у кінцевому консольному додатку голосової біометрії.

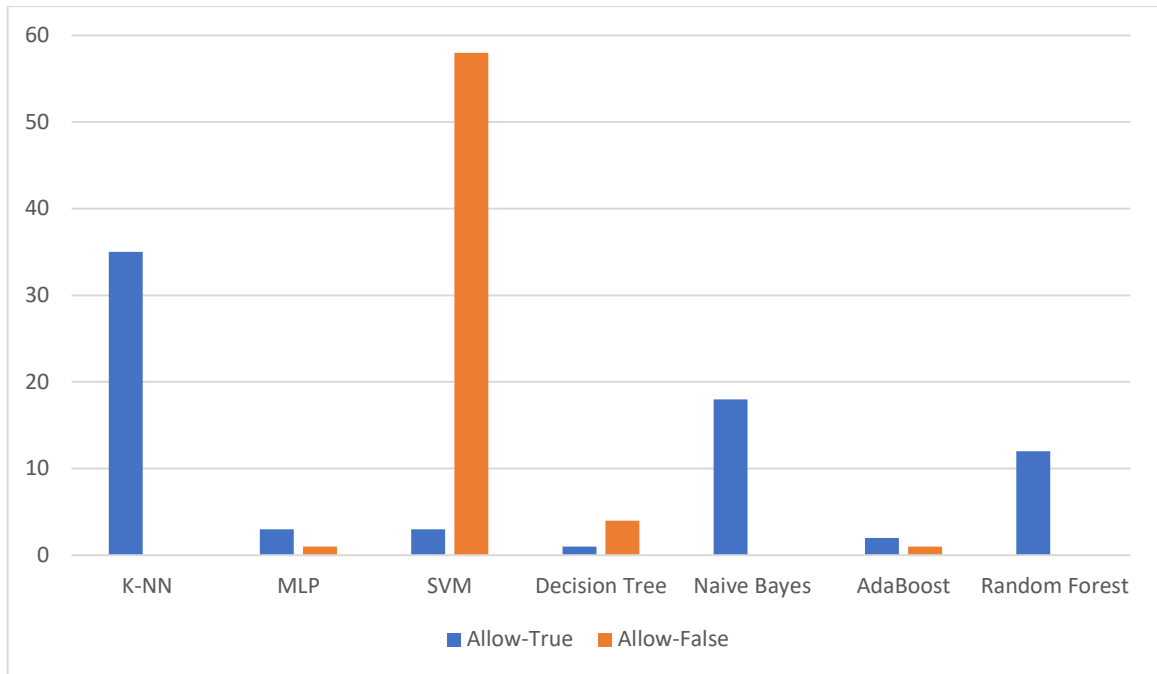


Рисунок 2.8. Порівняння точності семи різних класифікаторів по критеріям Allow-true (сині стовпчики; чим більше цей показник – тим вища точність класифікатора) та Allow-false (жовті стовпчики; чим менше цей показник – тим вища точність класифікатора)

2.4 Методи виявлення статичних викидів

Взагалі під поняттям Однокласової класифікації, або іншими словами Унарної класифікації розуміють процес навчання моделі об'єктами тільки одного класу. Це робиться для того, щоб потім немарковані об'єкти віднести або до цього класу, або до другого якогось невідомого класу (тобто виявити аномалію). У даному підрозділі буде розглянуто три методи унарної класифікації: LOF, One-class SVM та Isolation Forest.

2.4.1 Local Outlier Factor (Локальний коефіцієнт аномалії)

Алгоритм LOF (або Локальний коефіцієнт аномалії) був запропонований Маркусом М. Бренінгом, Гансом-Пітер Крігелем, Реймондом Т. та Йоргом

Сандером для знаходження аномальних точок у заданому наборі даних у 2000 році [17].

Метод спочатку навчається певним набором маркованих точок, після чого може віднести наступні немарковані точки до одного з двох класів: свій/чужий (аномальний).

LOF працює по принципу порівняння щільності досліджуваних точок з щільностями їх маркованих сусідів. Таким чином, якщо щільність даної точки значно менше за щільність її сусідів, то робиться висновок, що дана точка аномальна, тобто є чужа для навченого набору однорідних даних.

Основна ідея методу показана на рис. 2.9.

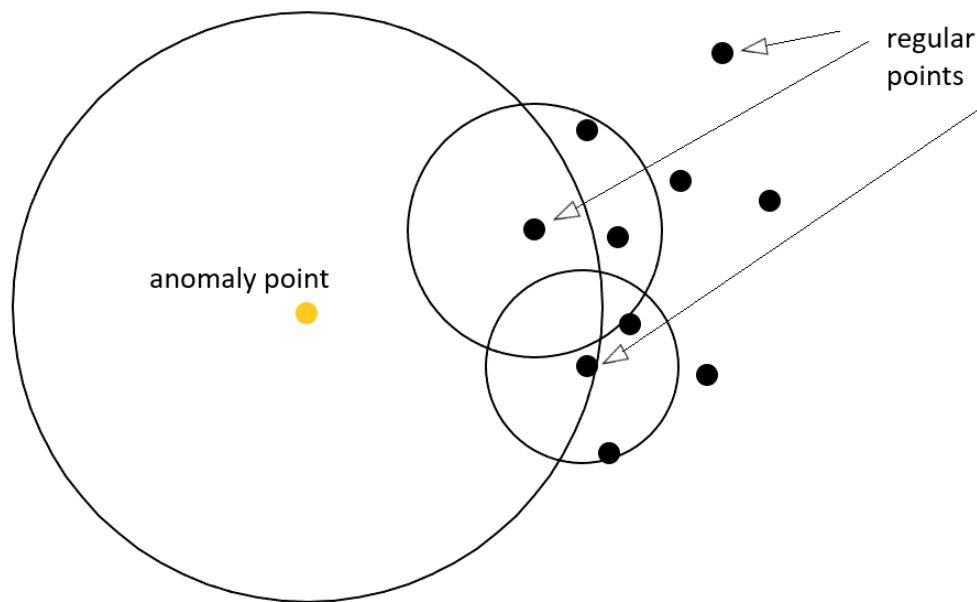


Рисунок 2.9. Основна ідея методу Локального коефіцієнту аномалії

Серед переваг LOF слід зазначити те, що він може спрацювати тоді, коли інші методи не спрацюють. Це спричинено тим, що прийняття рішення про аномальну точку залежить не просто від щільності досліджуваної точки, а і від щільності

сусідніх до неї точок. Таким чином, якщо взяти дві точки з однаковою власною щільністю, то одна із них може виявитись аномальною, а інша звичайною. Це трапляється тому, що в першому випадку щільність досліджуваної точки буде меншою за ту що у її сусідів, а в другому – більшою.

Слід зауважити, що в даній магістерській роботі саме цей метод було вибрано як кращий серед трьох: LOF, One-Class SVM, Isolation Forest. Однак, у цього алгоритму існує і ряд недоліків, найсуттєвішим з яких є те, що поріг при якому точку вже можна вважати аномальною може мінятися від одного набору даних до іншого.

2.4.2. One-class SVM (Однокласовий SVM)

One-class SVM (Однокласовий SVM) належить до методів унарної класифікації. Також цей метод називають Support Vector Data Description (SVDD).

Основна ідея методу полягає у знаходженні не гіперплощини, як у звичайному SVM, а сфери з мінімальним радіусом, яка б покрила всі точки, що були промарковані одним класом [18]. Дана ідея зображена графічно на рис. 2.10.

Цю ж ідею можна зобразити формулою 2.15.

$$\min_{R^2, c}(R^2): \|F(x_i) - c\|^2 < R^2, \quad \forall i = 1, 2, \dots, K. \quad (2.15)$$

Однак, при прямому застосуванні даного підходу, класифікатор буде досить чутливим до шуму. Через що, був розроблений нечіткий метод, в якому ця сфера нечітко задана.

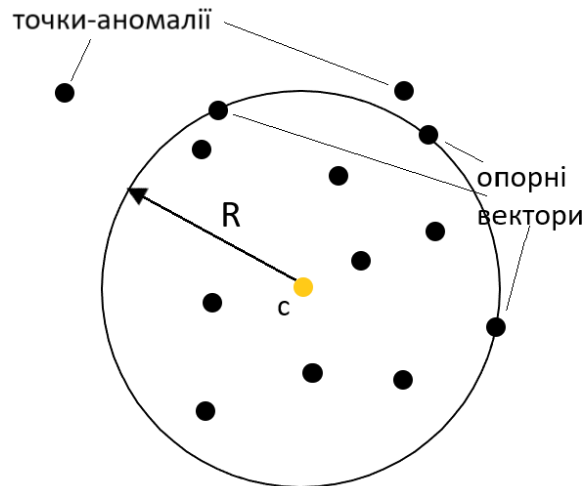


Рисунок 2.10. Основна ідея методу One-class SVM

2.4.3. Isolation Forest

Isolation Forest належить до ансамблевих методів унарної класифікації. Ансамблевий, тому що для однієї вибірки будується багато дерев, що утворюють ліс – ансамбль. В рамках даного підходу будуються ізоляційне дерево для кожного із вхідних образів. Довжина дерева є метрикою у функції прийняття рішення про аномалію. У аномальних образах дерева виходять значно коротшими. Якщо детальніше розглядати даний метод, то маємо наступне: ізоляційний ліс покроково “ізолює” образи, випадково обираючи координату вектора образа та значення по якому буде відбуватися ізоляція. Це значення обирається у межах допустимих значень ($[min, max]$) даної координати. Для кожного образу таку покрокову “ізоляцію” можна зобразити у вигляді дерева, у корні якого не ізолюваний образ, а листок якого – ізолюваний образ. Довжина дерева дорівнює кількості “ізоляційних” кроків. Як було вже сказано вище, для кожного образу будується таке дерево. У рамках всіх образів – отримуємо ліс. Для образів-аномалій, довжина дерев значно менша. Таким чином можна відсіяти точки аномалії від звичайних точок [9].

Висновки до розділу

У даному розділі був проведений огляд математичних методів розпізнавання голосового відбитку, а саме проаналізований алгоритм формування відбитку методом MFCCs Також було розглянуто класифікатори для розпізнавання відбитку, що наявні в бібліотеці аналізу даних scikit-learn, а саме: K-NN (K-Nearest Neighbours-К найближчих сусідів), MLP (Multilayer Perceptron – Багатошаровий перцептрон), SVM (Support Vector Machine – Вектор опроних векторів), Дерева ухвалення рішень, RFC (Random Forest Classifier - Випадковий ліс), алгоритм бустінгу AdaBoost, Наївний баєсів класифікатор, HMM (Hidden Markov Model – Прихована Марківська Модель). Також досліджені методи статичних викидів: Local Outlier Factor, однокласовий SVM, Isolation Forest.

З урахуванням проведеного огляду, аналізу та порівняння всіх методів та класифікаторів було прийнято рішення використовувати поєднання методу MFCCs та класифікатора К найближчих сусідів для розробки системи. Було описано загальну схему роботи методу MFCCs, що використовується для отримання голосового відбитку особи.

Об'єктом дослідження даної магістерської дисертації є розпізнавання голосу людини комп'ютером. Предмет дослідження – голосова біометрія, тобто голосове розпізнавання особи. Мета роботи полягає в створенні підсистеми голосового розпізнавання особи, що характеризувалася б невибагливістю до обчислювальних ресурсів та порівняно високою точністю роботи. З цією метою розроблено консольний додаток голосової біометрії.

3. ПРОЕКТУВАННЯ ПІДСИСТЕМИ ГОЛОСОВОЇ БІОМЕТРІЇ

3.1 Загальна структурна схема підсистеми голосової біометрії.

Даний підрозділ присвячений розробці загальної схеми підсистеми автентифікації особи по голосу та його модулів.

Як зазначалося у кінці першого розділу, кінцевою ціллю даної роботи є створення невибагливої до ресурсів, швидкої і до того ж досить точної системи голосової верифікації. Для цього спочатку було змодельовано її загальну структурну схему (рис. 3.1).

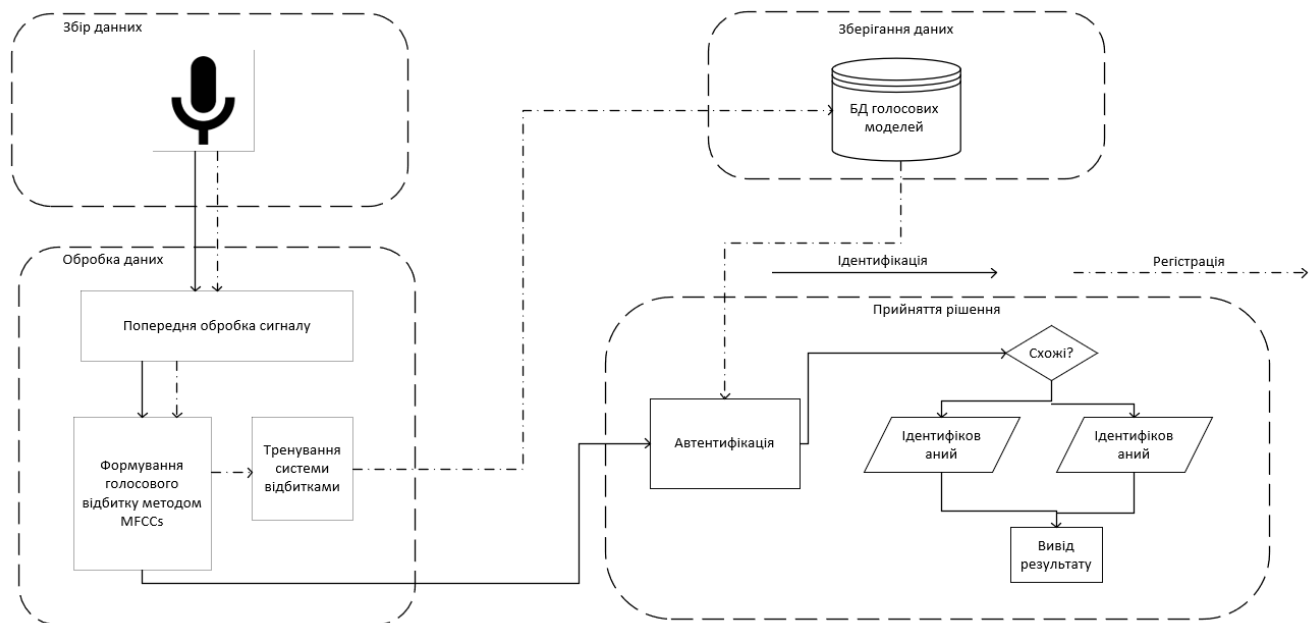


Рисунок 3.1. Загальна структурна схема спроектованої підсистеми

Далі варто описати кожен з основних модулів більш детально. Для початку розглянемо модуль, який відповідальний за навчання підсистеми голосом людини.

3.2 Схема модуля відповідального за навчання підсистеми голосом

Далі, так як і обіцялось в пункті 3.1, розглянемо більш детально кожен з модулів загальної схеми. Для початку розглянемо модуль, що відповідальний за тренування системи голосом людини.

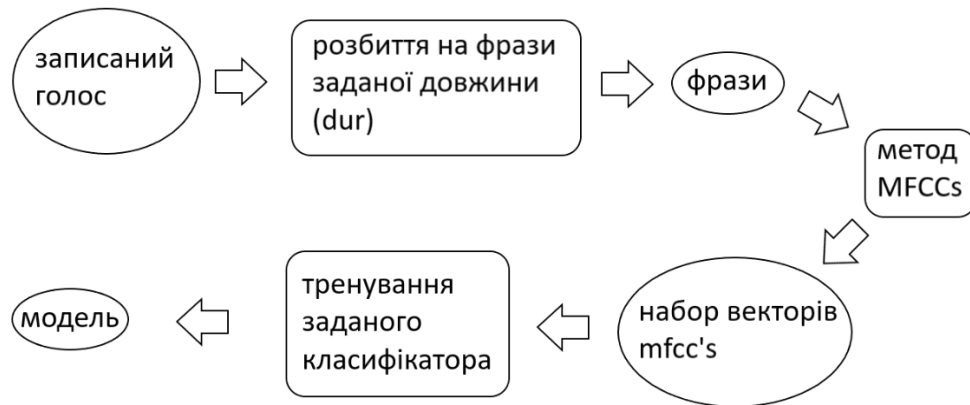


Рисунок 3.2. Опис модуля, який відповідальний за тренування підсистеми голосом людини.

Аналізуючи зображене на рис. 3.2, ми бачимо, що наш відбиток голосу особи ділиться на вирази певної довжини. З кожного виразу ми отримуємо відбиток голосу диктора, тобто вектор мел-кепстральних коефієнтів. Наступним етапом є впровадження отриманих голосових відбитків диктора у класифікатор. Методи k-nn, svm, mlr та інші можуть слугувати у якості класифікатора. Найвищий рейтинг серед названих методів отримав класифікатор KNN, саме тому його використання у даній роботі буде найдоцільнішим.

Завдяки роботі даного методу, ми зможемо використати модель на повну її силу на етапі аутентифікації диктора.

3.3 Схема модуля відповідального за автентифікацію особи.

Даний модуль більш детально показаний на рис. 3.3

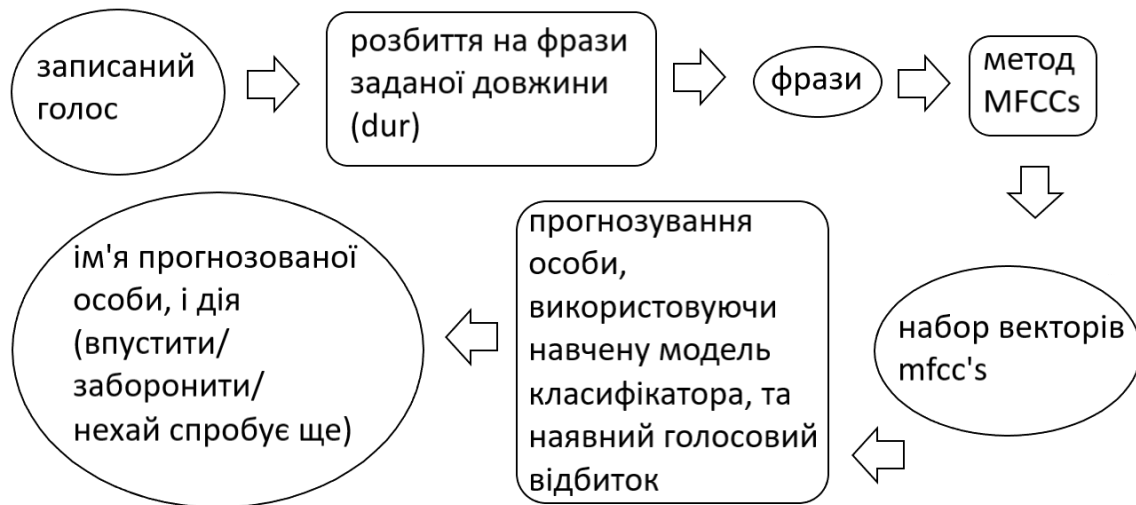


Рисунок 3.3. Модуль відповідальний за автентифікацію особи

На рис. 3.3. відображено, що частина другого модуля має певну ідентичність з модулем 1. Їх схожість полягає у ідентичності декількох етапів, таких як: запис голосу диктора, поділ на вирази певної довжини і відокремлення з кожного виразу відбитку голосу особи. Проте відмінність першого модуля полягає у використанні створеної моделі, за допомогою якою генерується індивідуальне ім'я об'єкта на основі голосового зразка. Програма співвідносить такі характеристики, як ймовірність прогнозу диктора з найвищим шансом ідентифікації до ймовірності іншою. Отже, коли ці співвідношення мають достатньо великий розрив між двома заданими характеристиками, то ми з більшою впевненістю можемо стверджувати, що об'єкт знаходиться у фазі ідентифікації.

Отже, при умові, якщо ми матимемо ім'я диктора у підсистемі, то зможемо досягти повної аутентифікації об'єкта та відповідно алгоритм дій, який підсистема матиме застосувати до ідентифікованої особи. Наприклад, це можуть бути такі дії, як: відхилення допуску, підтвердження допуску, тощо.

3.4 Опис роботи підсистеми

При проектуванні даного програмного продукту було прийнято рішення використати метод мел-частотних кепстральних коефіцієнтів (MFCC) для отримання значущих ознак у аудіоданих користувачів і алгоритм класифікації даних на основі моделей k-найближчих (KNN), так як MFCC порівняно з іншими методами володіє хорошим співвідношенням швидкості роботи до продуктивності, а KNN добре себе зарекомендував у задачі класифікації даних, які виділені із звукових висловлювань відрізняються не тільки варіаціями у вимові фрази, але і самими фразами. Іншими словами, даний програмний продукт є текстонезалежною підсистемою розпізнавання особистості користувача. Нижче описується її робота.

Покажемо на блок-схемі алгоритм роботи підсистеми голосової ідентифікації.

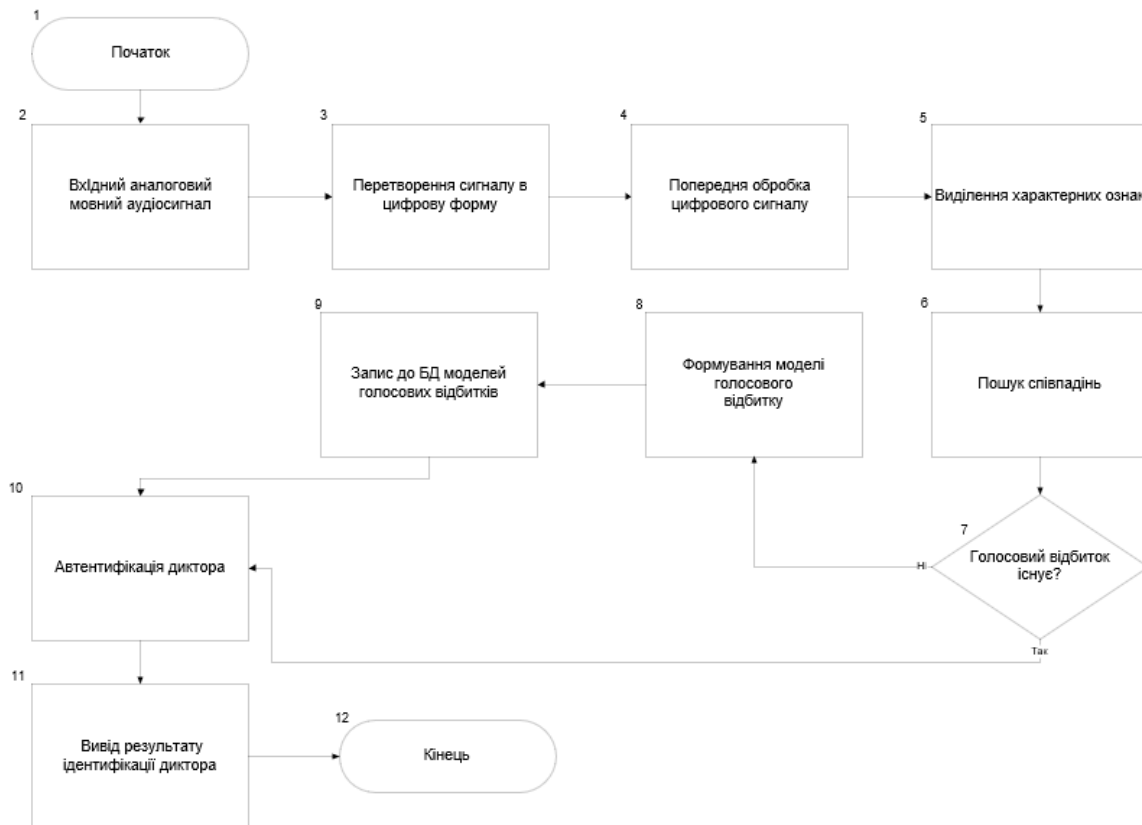


Рисунок 3.4. Блок-схема алгоритму роботи підсистеми

Як бачимо з блок-схеми спочатку на вхід подається аналоговий мовний сигнал, який в ході перетворень перетворюється в оброблений цифровий сигнал. Після чого відбувається перевірка на співпадіння записаного голосу вже з існуючими голосами в БД, та в разі його наявності в базі даних відбувається автентифікація диктора та вивід результату автентифікації. Якщо ж співпадіння не знайдено, то формується модель голосового відбитку та записується до БД задля проходження автентифікації в майбутньому.

Також розглянемо функціональну схему розроблюваної нами підсистеми (рис. 3.5)

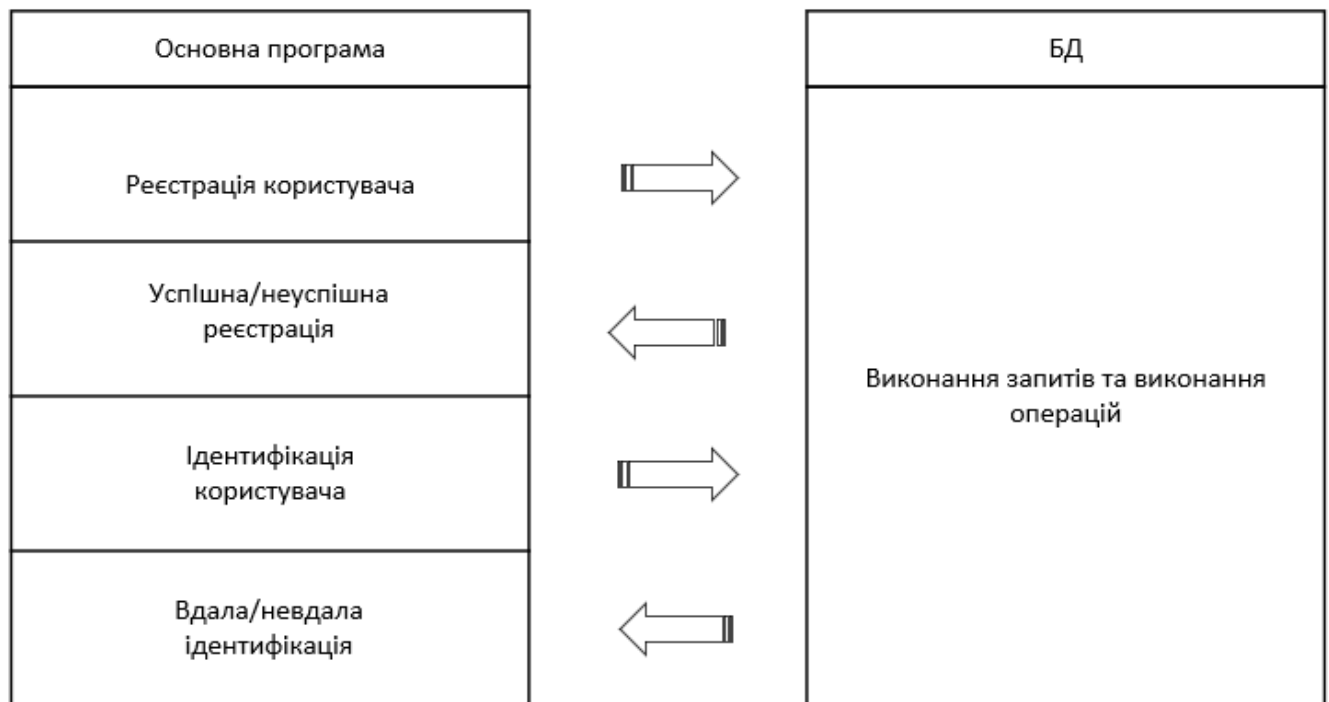


Рисунок 3.5. Функціональна схема підсистеми.

Висновки до розділу

У даному розділі було розроблено загальну структурну схему підсистеми для біометричної автентифікації по голосу. Було розроблено схему роботи методу MFCCs, що використовується для отримання голосового відбитку особи. Також

було розроблено блок-схему алгоритму роботи підсистеми, та її функціональну схему.

Також було більш детально описано модулі, що відповідальні за тренування системи та автентифікацію особи.

Виходячи с функціоналу було обрано інструменти для розробки ПЗ. Для розробки був обраний метод який з'єднав у собі математичні методи та методи машинного навчання. Такий підхід дозволяє отримувати майже ті самі результати, що і ШНМ, але не затрачуючи великі кошти на оренду потужних обчислювальних кластерів або купівлю дорогих та об'ємних серверів.

Об'єктом дослідження даної магістерської дисертації є розпізнавання голосу людини комп'ютером. Предмет дослідження – голосова біометрія, тобто голосове розпізнавання особи. Мета роботи полягає в створенні підсистеми голосового розпізнавання особи, що характеризувалася б невибагливістю до обчислювальних ресурсів та порівняно високою точністю роботи. З цією метою розроблено консольний додаток голосової біометрії.

4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Для розробки підсистеми була обрана мова програмування Python, так як вона є відкритою високорівневою мовою і підтримує велику кількість бібліотек для обробки та аналізу різноманітних даних. Розглянемо далі його функціональні можливості.

4.1 Мова програмування Python

Python – це мова програмування високого рівня, який створювався з метою збільшити читаність коду і підвищити продуктивність розробників. Створення мови почалося в кінці 1980-х Гвідо ван Россумом під впливом таких мов як ABC, C/C++, Java, Smalltalk та ін У 1991 році вихідний код був вперше опублікований на порталі «alt.sources» і з цієї події почався його активний розвиток. У процесі розвитку велика кількість ентузіастів цікавилася і робили свій внесок у поліпшення даної мови, що сприяло ще більшому зростанню популярності навіть серед непрофесіоналів і новачків. На даний момент підтримуються дві версії: 2.7 і 3.5 [20].

Головними рисами характеризують Python є:

- динамічна типізація – в процесі програмування не потрібно заздалегідь оголошувати тип змінної;
- підтримка об'єктно-орієнтованого програмування;
- проста установка великої кількості різноманітних модулів, які значно розширюють функціонал;
- легкість створення і використання власних модулів;
- підтримка кроссплатформенності;
- інтеграція з C/C++ і Java;
- більш простий для розуміння синтаксис в порівнянні з іншими високорівневими мовами;

- автоматичне управління пам'яттю.

4.2 Модулі та бібліотеки

Однією головних особливостей відрізняють пітона від інших мов є наявність великої кількості вільно поширюваних модулів і бібліотек, які значно розширюють функціональні можливості мови. Відповідно до поставленої задачі були обрані наступні модулі та бібліотеки:

- бібліотека `numpy`;
- `scikit-learn`;
- бібліотека `matplotlib`.

4.2.1 SciPy и SciKit-learn

У багатьох областях науково-дослідних і дослідно-конструкторських роботах потрібні просте у використанні і володіє великими можливостями програмне забезпечення. Для забезпечення даних потреб була створена SciPy – екосистема відкритого програмного забезпечення для математики, наукових досліджень та інженерних робіт. SciPy як екосистема ПЗ складається з наступних основних пакетів:

- NumPy;
- SciPy (як бібліотека);
- Matplotlib;
- IPython;
- SymPy;
- Pandas.

У стек пакетів SciPy входить однойменна бібліотека, яка займається обробкою даних для наукових та інженерних розрахунків. В її можливості і функції входять:

- константи (constants): фізичні константи і коефіцієнти для їх перерахунку;
- кластеризація (cluster): алгоритми кластеризації (векторне квантування, k-середні);
- пакет fftpack: алгоритми дискретного перетворення Фур'є;
- модуль «integrate»: інструменти для інтегрування;
- інтерполяція (interpolate): інструменти для інтерполяції даних (рис. 3.1);
- модуль «io»: введення і виведення різних даних;
- модуль «lib»: інтерфейси для роботи з зовнішніми бібліотеками;
- модуль лінійної алгебри "linalg": різні інструменти для роботи з лінійною алгеброю;
- модуль «misc»: різні утиліти загального призначення;
- модуль "ndimage": різні функції для роботи з багатовимірними зображеннями;
- алгоритми оптимізації («optimize»): різні алгоритми оптимізації в тому числі і лінійне програмування;
- обробка сигналу («signal»): інструменти для обробки сигналів;
- модуль «sparse»: розряджені матриці і алгоритми для роботи з ними;
- модуль «spatial»: просторові дані і алгоритми для роботи з ними, наприклад вирішальні дерева найближчий сусід;
- модуль "special": спеціальні функції;
- статистичні функції ("stats»);

- модуль «weave»: інструменти для впровадження коду C / C++ в Python у вигляді багаторядкових коментарів.

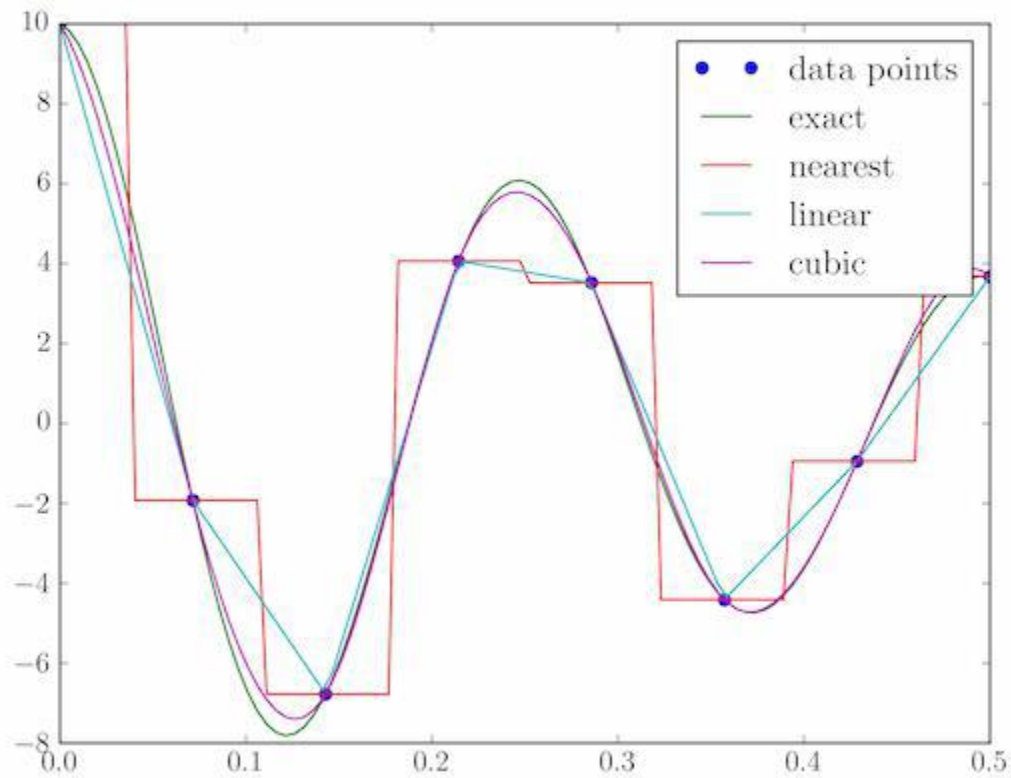


Рисунок 4.1. Різні методи інтерполяції даних в `scipy.interpolate`

Для ще більшого розширення можливостей екосистеми програмного забезпечення SciPy був створений набір додаткових пакетів під назвою SciKits (SciPy Toolkits). У набір входять різні модулі: інструменти для створення і роботи з нейронними мережами, машинне навчання, нечітка логіка, робота з зображеннями, розрахунок аеродинаміки, геодезія і багато інших. Для аналізу оброблених даних було вирішено використовувати модуль SciKit-learn, який включає в себе засоби для роботи з великою кількістю даних і їх аналізу, машинне навчання і комп'ютерного бачення [21].

Нижче представлений список актуальних завдань в сфері обробки даних завдань, які SciKit-learn здатний вирішити:

- кластеризація (Clustering): створення та присвоєння класів об'єктам на нерозмічених даних;
- класифікація (Classification): визначення належності об'єкта до вже заданих класів (рис. 3.2);
- перехресна перевірка (Cross Validation): методи оцінки ефективності роботи алгоритмів за допомогою незалежних даних;
- набори даних (Datasets): вбудовані набори даних для тестування алгоритмів;
- скорочення розмірності (dimensionality Reduction): для виділення більш значущих ознак і даних;
- алгоритмічні композиції (Ensemble Methods): для комбінування передбачень декількох моделей;
- витяг ознак (Feature Extraction): визначення атрибутів в зображеннях і текстових даних;
- відбір ознак (Feature Selection): виявлення найбільш значущих ознак для побудови моделей;
- оптимізація параметрів алгоритму (Parameter Tuning): для отримання максимально ефективної віддачі від моделі;
- множинне навчання (Manifold Learning): для нелінійного скорочення розмірності даних;
- алгоритми навчання з учителем(Supervised Models): велика кількість алгоритмів для навчання на тестових даних.

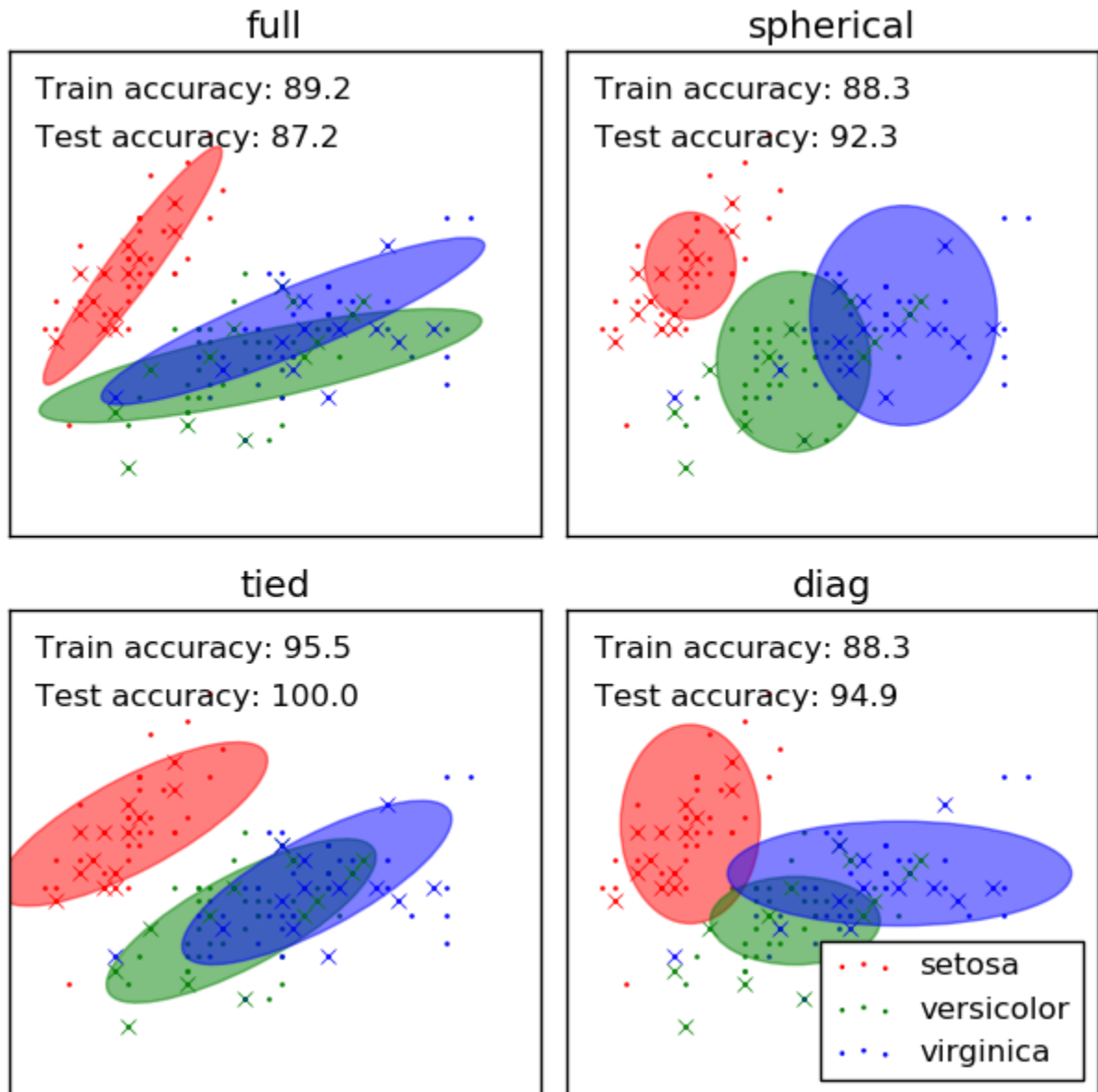


Рисунок 4.2 Класифікація трьох видів ірису за допомогою чотирьох видів GMM

4.2.2 Matplotlib

Matplotlib – це бібліотека візуалізації даних у двовірному та тривимірному вигляді. З її допомогою розробник може відображати в різному вигляді всілякі дані. Дана бібліотека здобула велику популярність у зв'язку з тим, що вона забезпечує легкість в програмуванні візуалізації даних і великі функціональні можливості, які дозволяють підготувати дані до публікації в наукових роботах. Зв'язка NumPy і

Matplotlib є гідною відповіддю таким пакетам математичного моделювання як MATLAB і Mathematica. На рис. 4.3 наведено приклад діаграми, що візуалізується в Matplotlib [22].

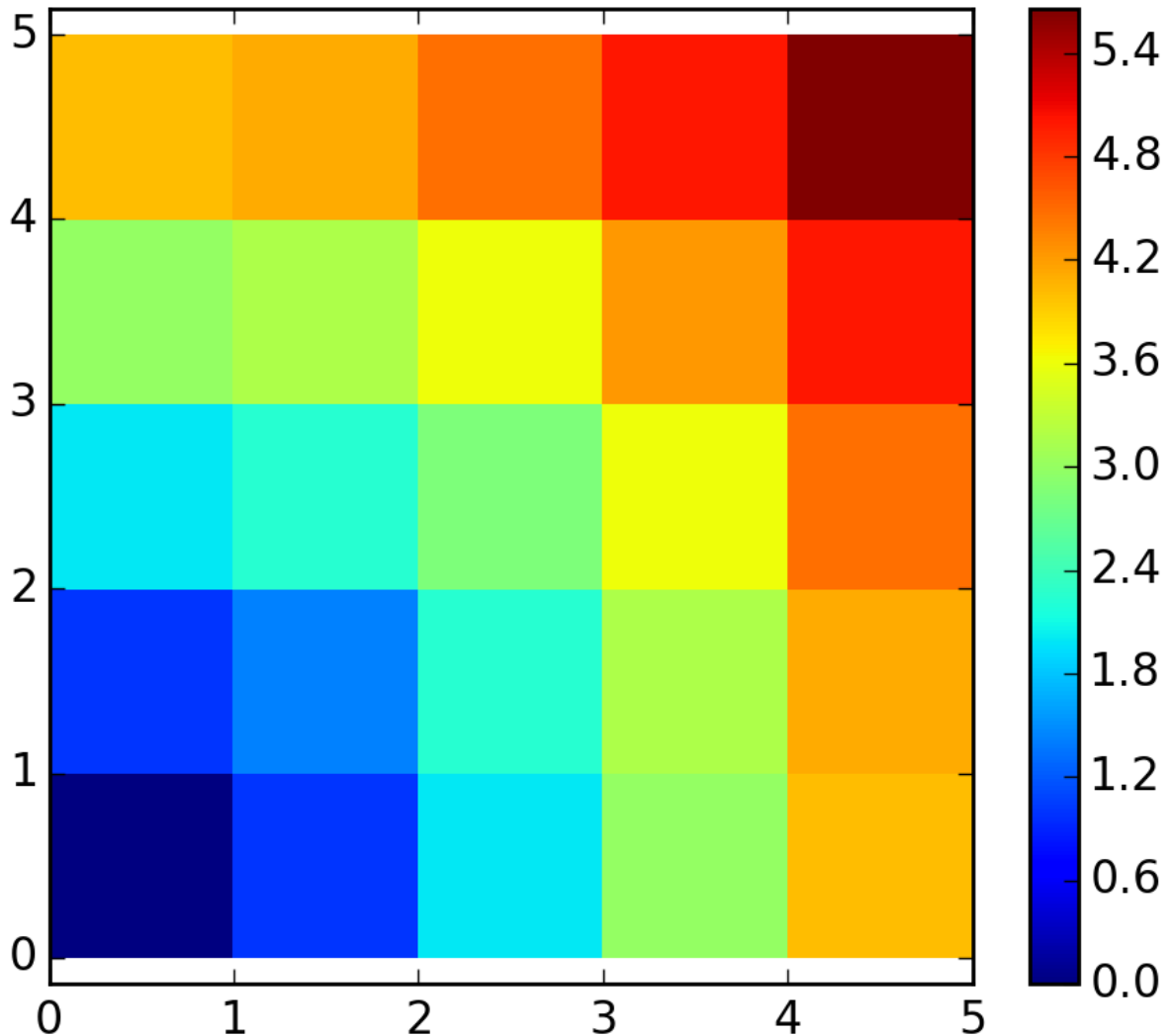


Рисунок 4.3. Приклад графіка Matplotlib

Види діаграм підтримувані Matplotlib:

- звичайні графіки ("line plot»);
- діаграми розкиду ("scatter plot»);
- стовпчасті діаграми ("bar chart»);

- гістограми ("histogram»);
- кругові діаграми ("pie chart»);
- стовбур-лист діаграми ("stem plot»);
- контурні графіки ("contour plot»);
- поля градієнтів ("quiver»);
- спектрограми («spectrogram").

4.2.3 NumPy

На ряду з Matplotlib, NumPy є однією з найбільш використовуваних бібліотек при програмуванні на Python. Її інструменти, функції та об'єкти застосовуються практично в кожному виконуваному скрипті. Основна спеціалізація бібліотеки – високоефективна робота з багатовимірними масивами. Нижче перераховано основний функціонал даної бібліотеки:

- багатовимірні масиви;
- функції для обробки даних n-мірних масивів;
- основні функції лінійної алгебри;
- базові перетворення Фур'є;
- функції для роботи з випадковими величинами;
- інструменти для інтеграції коду на Fortran;
- інструменти для інтеграції коду на C / C++.

4.3 Опис роботи консольного додатку голосової біометрії

Перед показом прикладів роботи консольного додатку розглянемо блок схему-схему алгоритму його роботи

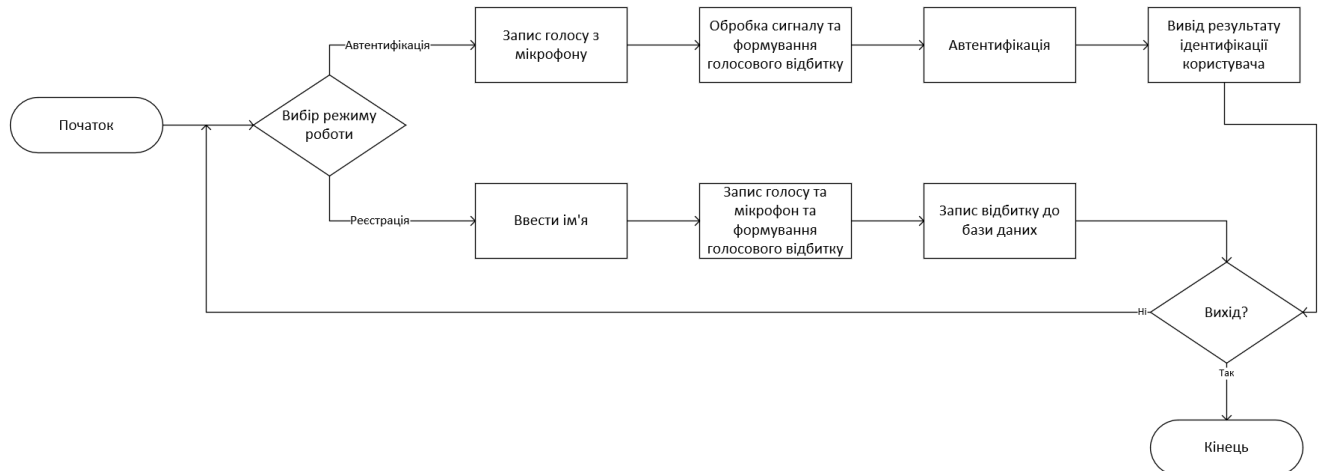


Рисунок 4.4. Блок-схема алгоритму роботи консольного додатку

Тепер більш детально, за допомогою скріншотів, покажемо, як функціонує програма. Перейдемо до етапу, коли підсистема буде навчатися на прикладі власного голосу. Отже, першим кроком у цьому процесі є запуск файлу writeToDb.py. Далі необхідно ідентифікуватися, тобто написати власне ім'я і підтвердити, шляхом натискання на клавішу «Enter». На рис. 4.5 зображено вікно, яке з'явиться після проходження зазначених маніпуляцій. Фразу, що зображена на рисунку необхідно повторювати поки ми не отримаємо повідомлення з текстом «Фраза записана!», як показано на рис. 4.6.

```

введіть своє ім'я (на англійській): dima
=====
прочитайте фразу в мікрофон поки не вісвітиться повідомлення (20 сек):
=====
мисливців-збирачів. Археологи знайшли близько 800 пам'яток цих людей в Україні і виділяють їх у закарпатську, дністровську
, волинську,
середньодніпровську та степову групи.[8] Серед них особливо виділяють природний останець Кам'яна Могила, який став культов
им центром
кроманьйонців степової зони[9].

Понад 10 тисяч років тому відбувся перехід від палеоліту до мезоліту, який збігся з таненням льодовика та початком нової г
еологічної
доби – голоцену. Загальне потепління сприяло збільшенню кількості населення[10]. Проте криза привласнюваного мезолітичного
господарства
поступово змусила людей приступити до відтворювальних форм: рільництва і скотарства. Це сприяло винаходу кераміки. Настала
нова доба
неоліту, яка тривала протягом 6–4 тисячоліть до н. е. Стабілізувався ландшафтний поділ України на лісову, лісостепову і ст
епову зони,
утворився гумусний покрив землі. Неолітичні культури України формувалися під впливом досягнень осередків Близького Сходу,
які імпортувалися
переважно через Балканський півострів і Подунав'я[11].
  
```

Рисунок 4.5. Консоль програми на етапі навчання власним голосом (під час надиктовки тексту)


```
=====
=====
=====
=====
=====ФРАЗА ЗАПИСАНА!=====
=====
=====
=====
mfccs було записано в .csv
-
```

Рисунок 4.6. Консоль програми на етапі навчання власним голосом (після надиктовки тексту)

Далі необхідно повторити вищезазначені маніпуляції з іншими дикторами, аби у підсистемі було більше двох голосових зразків. Після цього етапу можна сміливо переходити до аутентифікації. Цей етап має спільні ознаки з етапом запису голосового відбитку у підсистему. Перш за все ми запускаємо програму authentication.py і також повторюємо текстове повідомлення, яке буде відображатися на екрані. Один з варіантів тексту зображений на рис. 4.7. Програма аналізує голос протягом достатньо короткого проміжку часу, а саме 6 секунд.

```
=====
прочитайте фразу в мікрофон поки не вісвітиться повідомлення (6 сек):
=====
Пантикапей (біля сучасної Керчі), Феодосія. Ці держави були демократичними або аристократичними за устроєм. У цих містах панували рабовласницькі відносини. Головним джерелом постачання рабів був військовий полон, народження від рабині чи купівля на невідланих ринках. Права у полісі мали лише повнолітні чоловіки-греки, уродженці міста. Колонія складалася власне з поліса та сільськогосподарських округів. Міста оснащено водогоном та водостоком, були поширені ремесла й торгівля. Міста карбували власні монети.

Боспорська держава (V ст. до н. е. – IV ст. н. е.) займала територію сучасного Керченського та Таманського півострова. До складу царства увійшли такі міста, як Феодосія, Фанагорія, а столицею був Пантикапей. Спершу це був союз полісів, які мали певну автономію, та згодом це об'єднання перетворилося на абсолютну монархію. Економіка цього царства була побудована на сільському господарстві та торгівлі з Афінами, куди вивозили до 5 млн. пудів зерна. У I ст. до н. е. відбулося об'єднання під владою понтійського царя Мітрідата VI більшості Північного Причорномор'я, але він зазнав поразки від римлян.
```

Рисунок 4.7. Консоль програми на етапі проходження аутентифікації власним голосом під час надиктовки тексту

Потім на екрані з'явиться повідомлення з текстом «Фраза записана!», а також результатом аутентифікації і алгоритмом дій, який підсистема матиме застосувати до встановленої особи. Розшифруємо три основні дії підсистеми, що зв'язані з

програмою: Allow – доступ дозволено, Deny – доступ відхилено, Try More – спробувати ще. Наочний приклад ви можете побачити на рис. 4.8.

```
=====
=====
=====
=====
=====ФРАЗА ЗАПИСАНА!=====
=====
=====
=====

=====
=====results:=====
=====

DURATION: 4.5
C:\Users\Windows\AppData\Local\Programs\Python\Python37\lib\site-packages\sklearn\neighbors\lof.py:236: FutureWarning: de
fault contamination parameter 0.1 will change in version 0.22 to "auto". This will change the predict method behavior.
  FutureWarning)
{-1: 150, 1: 292}
NOVELTY_RELATION: 1.947
RELATION: 3.585
individual: dima
action: ALLOW
```

Рисунок 4.8. Консоль програми на етапі проходження аутентифікації власним голосом (після надиктовки тексту)

Враховуючи результати дослідження описаного вище, ми з впевненістю можемо стверджувати, що головна ціль роботи була досягнута успішно. Ми створили достатньо швидку підсистему аналізу голосових зразків, яка окрім своєї простоти, має достатньо високий рівень точності.

Завдяки цим критеріям, підсистема може слугувати ефективним інструментом у багатьох галузях. Також підсистема може використовуватися, як для задоволення дрібних проблем, таких як розблокування будь-якого пристрою, так і для реалізації на серверах тих чи інших підприємств. А враховуючи економію при використанні цієї підсистеми, вона має майбутнє і як інструмент для допомоги малому бізнесу.

Висновки до розділу

У даному розділі було обрано інструментарій, що використовується для розробки ПЗ. Мовою програмування, на якій написано консольний додаток, було обрано Python, який задовольняє всі потреби цієї розробки. Також було розглянуто основні модулі та бібліотеки, які дозволяють вирішити поставлену задачу.

У розділі було розроблено загальну блок-схему алгоритму роботи кінцевого консольного додатку голосової біометрії.

Також були подані приклади роботи розробленого консольного додатку у вигляді скріншотів.

Основна характеристика додатку це час затрачений на обробку голосових сигналів і вона має два показники: 22 секунди було затрачено під час навчання підсистеми, при тому, що промова складала 20 секунд, і 6 секунд запису під час аутентифікації, при тому, що сама обробка отриманого голосового відбитку складала всього 2 секунди.

Отже, враховуючи результати дослідження описаного вище, ми з впевненістю можемо стверджувати, що головна ціль роботи була досягнута успішно. Ми створили достатньо швидку підсистему аналізу голосових зразків, яка окрім своєї простоти, має достатньо високий рівень точності.

Завдяки цим критеріям, підсистема може слугувати ефективним інструментом у багатьох галузях. Також підсистема може використовуватися, як для задоволення дрібних проблем, таких як розблокування будь-якого пристрою, так і для реалізації на серверах тих чи інших підприємств. А враховуючи економію при використанні цієї підсистеми, вона має майбутнє і як інструмент для допомоги малому бізнесу.

5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

У даному розділі буде розглянуто ключові особливості розробленої підсистеми як майбутнього стартап-проекту. Проект розглядатиметься як підсистема голосового розпізнавання людини.

5.1 Опис ідеї проекту

Спочатку проаналізуємо та подамо у вигляді таблиці зміст ідеї стартап-проекту, можливі напрямки застосування та основні вигоди, які може отримати користувач товару. Ці характеристики стартап-проекту зображено в таблиці 5.1.

Таблиця 5.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Програмний додаток для аутентифікації людини по голосу.	1. Застосування як підсистеми голосової біометрії на веб-сайтах.	Можливість швидко отримати доступ до приватної частини веб-сайту і т. ін. Мінімальні витрати на апаратне забезпечення (сервер).
	2. Застосування як підсистеми голосової біометрії на смартфонах.	Додаткова підсистема аутентифікації на смартфоні.

Тепер зробимо аналіз потенційних техніко-економічних переваг ідеї порівняно із пропозиціями конкурентів. Результати аналізу зображено в таблиці 5.2.

Таблиця 5.2 Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	Товари/концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Конкурент 1	Конкурент 2			

Продовження таблиці 5.2

1.	Ціна	2000\$/ рік	4000\$/ рік	5000\$/ рік			+
2.	Прибутки	6000\$/ рік	40000\$/ рік	20000\$/ рік	+		
3.	Контроль якості	Аналі-тики та прог-рамісти	Аналі-тики, прог-рамісти та деякі клієнти	Прог-рамісти			+
4.	Динаміка галузі	Швидка	Пові-льна	Швид-ка		+	
5.	Постійні витрати	1000\$/ рік	20000\$/ рік	15000\$/ рік			+
6.	Змінні витрати	1000\$ -2000\$/ рік	1000\$ - 2000\$/ рік	2000\$ - 5000\$/ рік			+
7.	Патенти на продукти	Немає	Патент на кож-ний проект	Декі-лька патен-тів на винахід	+		
8.	Гнучкі ціни	Ціна варію-ється з року в рік	Ціна варію-ється з року в рік	Ціна єдина			+

Закінчення таблиці 5.2

9.	Законо- давчі обмеження	Обмеження на використання приватних даних (голоса) людини — GDPR	Немає	Обме- ження на кіль- кість розро- бників			
----	-------------------------------	---	-------	---	--	--	--

5.2 Технологічний аудит ідеї проекту

Визначимо технологічну здійсненність ідеї проекту за допомогою аналізу таких складових, як технології, за якою буде виготовлено товар згідно ідеї проекту, існування таких технологій, чи їх необхідно розробити / доробити, доступність таких технологій авторам проекту. Результати даного аналізу зображено в таблиці 5.3.

Таблиця 5.3 Технологічна здійсненність ідеї проекту

Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
Програмний додаток для ідентифікації людини по голосу.	Технологія проектування та розробки системи для вилучення ознак з голосу людини MFCCs python_speech_features [24].	Так	Дані технології доступні, але разом з ними необхідно використати подальші методи (класифікатори), зادля збільшення точності ідентифікації.

Закінчення таблиці 5.3

	Технологія проектування та розробки класифікаторів sklearn [21].	Так	Дані технології доступні, однак, штатного функціоналу все ще не вистачає для вирішення поставлених задач.
	Технологія здійснення порівняльного аналізу sklearn. .neighbors.LocalOutlierFactor	Так	Дані технології доступні.
Обрана технологія реалізації ідеї проекту: технологія вилучення ознак з голосу (python-speech-features [24]), класифікації та порівняння (sklearn [21]).			

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку. Результати даного аналізу зображено в таблиці 5.4.

Таблиця 5.4 Попередня характеристика потенційного ринку стартап-проекту

<i>№ n/n</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	300 000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Висока точність розпізнавання, швидкодія, невибагливість до ресурсів
5	Специфічні вимоги до стандартизації та сертифікації	GDRP

Закінчення таблиці 5.4

6	Середня норма рентабельності в галузі (або по ринку), %	65
---	---	----

Таким чином, за попереднім оцінюванням, ринок є привабливим для входження.

Надалі визначимо потенційні групи клієнтів, їх характеристики, та сформуємо орієнтовний перелік вимог до товару для кожної групи. Ці дані зображено в таблиці 5.5.

Таблиця 5.5 Характеристика потенційних клієнтів стартап-проекту

<i>№ n/n</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1	Голосова біометрія у веб-додатках	Малий бізнес	Малому бізнесу буде потрібно мати панель для адміністрації наявних в системі акаунтів. При цьому заплативши мінімум грошей на сервери.	Клієнти прагнуть збереженню їх особистих даних (голосу). Також прагнуть точності та швидкодії від системи.
2	Голосова біометрія у мобільних додатках	Малий бізнес	У малого бізнесу може виникнути потреба у інтеграції даної системи голосової біометрії з іншими системами аутентифікації	Клієнти прагнуть збереженню їх особистих даних (голосу). Також прагнуть точності та швидкодії від системи.

Після визначення потенційних груп клієнтів проведемо аналіз ринкового середовища: складемо таблиці факторів, що перешкоджають ринковому впровадженню проекту (таблиця 5.6), та факторів, що йому сприяють (таблиця 5.7).

Таблиця 5.6 Фактори загроз

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Відсутність попиту	Бізнес може не оцінити переваги продукту, або ж у цілому відмовитися від голосової біометрії у своїх кол-центрах	Акцентувати увагу на клієнтах, що вже скористалися продуктом, якщо такі є, навести інфографіку результативності (очікувану), запропонувати знижку потенційному клієнту в рамках тендеру.
2	Неточне розпізнавання	Дефекти та особливості мовлення певних осіб можуть привести до зниження точності системи голосової біометрії	Розробка і випуск оновлення ядра системи, де виправлена ця проблема.

Таблиця 5.7 Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1	Кобрендінг	Пропозиція від певної компанії, що спеціалізується на системах голосової біометрії, розробити спільний продукт	Виділення частини штату на реалізацію кобрендінгу, кооперація власними розробками між компаніями

Надалі проведемо аналіз пропозиції: визначимо загальні риси конкуренції на ринку. Результати даного аналізу зображені в таблиці 5.8.

Таблиця 5.8 Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Чиста конкуренція	Гравці ринку не мають явних переваг один над одним	Маркетинг (пояснення ідеї дешевої вартості, підтримки та гарної точності додатку)
2. Регіональна конкуренція	Гравці ринку – інтернаціональні підприємства	Маркетинг (пояснення ідеї дешевої вартості, підтримки та гарної точності додатку)
3. Внутрішньогалузева конкуренція	Гравці ринку знаходяться в одній галузі – розробці ПЗ	Маркетинг (пояснення ідеї дешевої вартості, підтримки та гарної точності додатку)
4. Товарно-видова конкуренція	Усі продукти гравців ринку мають одне призначення	Розробка найбільш інтуїтивного інтерфейсу Маркетинг (пояснення ідеї дешевої вартості, підтримки та гарної точності додатку)
5. Конкурентні переваги нецінові	Продукти відрізняються гнучкістю, функціоналом (незначно) і надійністю.	Маркетинг (пояснення ідеї дешевої вартості, підтримки та гарної точності додатку)
6. Марочна конкуренція	Значна увага приділяється бренду, що розробив продукт	Кобрендінг

Тепер визначимо та обґрунтуємо фактори конкурентоспроможності, які зображені в таблиці 5.9.

Таблиця 5.9 Обґрунтування факторів конкурентоспроможності

№ n/n	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Іноваційність	Продукт представляє реалізацію голосової біометрії – одного з напрямків Штучного Інтелекту
2	Невибагливість до апаратних ресурсів (серверів). А отже дешевизна апаратних ресурсів, потрібних для нашої системи	В продукті використане поєднання класичних методів голосового розпізнавання та методів Машинного навчання
3	Швидкодія	В продукті використане поєднання класичних методів голосового розпізнавання та методів Машинного навчання
4	Точність	Продукт має високу точність аутентифікації, яка може бути порівняна з системами більш високого класу
5	Юридична перевага	Відсутня потреба продукту у Big Data, що дає можливість зменшити пов'язаний з ним юридичний ризик
6	Інтеграція	Продукт може бути використаний в будь-якому веб-сайті захищеному протоколом ssl. Продукт не потребує придбання спеціалізованого апаратного забезпечення

Таблиця 5.10 Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальни- ки	Клієнти	Товари- замінники
	Динаміка галузі, продуктова лінія, бар'єри проникнення	Наявність товарних знаків, доступ до ресурсів, патенти на продукти	Концентрація постачальників, диференціація витрат	Рівень чутливості до зміни цін, прибутки, зворотній зв'язок	Ціна, лояльність споживачів

Закінчення таблиці 5.10

Висновки:	Конкуренція не є інтенсивною, адже конкурентів мало (або немає взагалі).	Для входу на ринок необхідно створити товарний знак та написати бета-версію програмного продукту.	Постачальники не диктують умови роботи на ринку, бо програмному продукту не потрібно постачання.	Клієнти диктують умови роботи на ринку, бо вони є єдиним джерелом прибутку компанії.	При наявності товарів замінників необхідно буде зменшувати ціну програмного продукту чи створювати ПЗ для інших технічних систем.
-----------	--	---	--	--	---

За визначеними факторами конкурентоспроможності проведемо аналіз сильних та слабких сторін стартап-проекту. Результати даного аналізу зображено в таблиці 5.11.

Таблиця 5.11 Порівняльний аналіз сильних та слабких сторін системи «BioM»

№ n/n	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з BioM						
			-3	-2	-1	0	+1	+2	+3
1	Іноваційність	15					*		
2	Невибагливість до ресурсів	18		*					
3	Швидкодія	18			*				
4	Точність	10						*	
5	Юридична перевага	14			*				

Тепер проведемо SWOT-аналіз на основі виділених загроз і можливостей, та сильних і слабких сторін проекту. SWOT-матриця зображено в таблиці 5.12.

Таблиця 5.12 SWOT-аналіз стартап-проекту

Сильні сторони: іноваційність, невибагливість до обчислювальних ресурсів, швидкодія, точність, юридична перевага	Слабкі сторони: Немає належного досвіду у веденні бізнесу
Можливості: Кобрендинг	Загрози: Відсутність попиту, неточність розпізнавання, порушення прав конфіденційності споживачів (GDRP)

На основі SWOT-аналізу розробимо альтернативи ринкової поведінки для виведення стартап-проекту на ринок та орієнтований оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок. Дані альтернативи зображено в таблиці 5.13.

Таблиця 5.13 Альтернативи ринкового впровадження стартап-проекту

<i>№ n/n</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Реалізація можливості використання системи не тільки на веб-сайтах, а й телефонних додатках	Середня	18 місяців
2	Створення системи емоційного розпізнавання людини	Висока	22 місяці
3	Розробка MVP	Висока	12 місяців

Серед даних альтернатив було обрано третю альтернативу, адже строки її реалізації найменші та є ймовірність отримання ресурсів.

5.4 Розроблення ринкової стратегії проекту

Для розроблення ринкової стратегії першим кроком необхідно описати цільові групи потенційних споживачів, які можна побачити в таблиці 5.14.

Таблиця 5.14 Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1.	Малий бізнес	Готові	2-3 підприємств в рік	Слабка	Середня
2.	Середній бізнес	Середня	1-2 підприємств в рік	Велика	Велика
3.	Великий бізнес	Мала	1 заклад в рік	Велика	Велика
Було обрано цільову групу підприємств групи малого бізнесу.					

Для роботи в обраних сегментах ринку необхідно сформулювати базову стратегію розвитку, якуображено в таблиці 5.15.

Таблиця 5.15 Визначення базової стратегії розвитку

Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Концентрація на потребах одного цільового сегменту – веб-сайтах.	Створений продукт є інноваційним дешевим початково та дешевим у використанні	Стратегія спеціалізації.

Наступним кроком є вибір стратегії конкурентної поведінки, яку зображено в таблиці 5.16.

Таблиця 5.16 Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
Так.	Компанія буде шукати нових споживачів	Компанія буде копіювати конкурентів.	Стратегія заняття ніші.

Тепер розробимо стратегію позиціонування, що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект. Її зображено в таблиці 5.17.

Таблиця 5.17 Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформулювати комплексну позицію власного проєдипломнікту (три ключових)
Розпізнавання особи за голосом має бути точним. Система є дешевою початково і у використанні Система є швидкою	Проведення крупних оновлень (оптимізація розрахунків), постійний зворотній зв'язок від клієнтів.	Товар є іноваційним та дешевим при покупці та у використанні порівняно з альтернативами.	Іноваційний, дешевий, невибагливий до обчислювальних ресурсів точний, зручний, програма працює в режимі онлайн.

5.5 Розроблення маркетингової програми стартап-проекту

Сформуємо маркетингову концепцію товару, який отримає споживач. В таблиці 5.18 зображено результати попереднього аналізу конкурентоспроможності товару.

Таблиця 5.18 Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1.	Прийнятна початкова ціна	Прийнятна початкова ціна, в порівнянні з альтернативами	Доступність для компаній з невеликим капіталом.
2.	Прийнятна вартість обслуговування	Невибагливість до апаратних ресурсів клієнта	Можливість встановлення на пристрої з обмеженою потужністю
3.	Іноваційність	Продукт належить до голосової біометрії - напрямку Штучного Інтелекту	Були використані перевірені бібліотеки машинного навчання
4.	Швидкість роботи	Швидкість аутентифікації	Можливість роботи в режимі онлайн
5.	Точне розпізнавання особи.	Точне розпізнавання особи за її голосом.	

Надалі розробимо трирівневу маркетингову модель товару: уточнимо ідею продукту, його фізичні складові, особливості процесу його надання. Дана модель зображена в таблиці 5.19.

Таблиця 5.19 Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Програмний продукт – система голосової біометрії, яка надає користувачу додаткову зручну альтернативу звичній нудній аутентифікації по пароллю.
II. Товар у реальному виконанні	Властивості / характеристики: 1. Можливість зареєструватися в систему за допомогою голосу 2. Можливість пройти аутентифікацію за допомогою голосу особи 3. Можливість пройти аутентифікацію іншим, альтернативним, способом
	Якість: програмний продукт пройшов всі етапи тестування та готовий до використання.
	Файл з розширенням “.ру”, віртуальне середовище.
	Марка: назва організації-розробника «YG», назва товару «BioM».
III. Товар із підкріпленням	Спеціаліст із впровадження встановлює ПЗ.
	Відділ розробки підтримує життєдіяльність ПЗ.
Захист програмного продукту буде організовано за допомогою ноу-хау.	

Тепер визначимо цінові межі, якими необхідно керуватись при встановленні ціни на потенційний товар, яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів. Аналіз проводився експертним методом і його результати зображено в таблиці 5.20.

Таблиця 5.20 Визначення меж встановлення цін

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
3000-5000 \$/рік	5000-6000 \$/рік	12000-50000 \$/рік	Нижня межа – 2000 \$/рік, верхня межа - 3000 \$/рік

Надалі визначимо оптимальну систему збуту, в межах якого приймається рішення. Дану систему зображено в таблиці 5.21.

Таблиця 5.21 Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина оптимізація каналу збуту	Оптимальна система збуту
Клієнт виплачує гроші на рік, тоді до нього приходять спеціаліст із впровадження інформаційних систем і встановлює ПЗ на комп'ютер клієнта.	Встановити програмний продукт на комп'ютери клієнтів.	Один посередник – спеціаліст по впровадженню інформаційних систем.	Канал збуту одного рівня.

Тепер розробимо концепцію маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів. Дану концепцію зображено в таблиці 5.22.

Таблиця 5.22 Концепція маркетингових комунікацій

Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
---------------------------------------	--	--	----------------------------------	--------------------------------

Закінчення таблиці 5.22

Клієнт намагається знайти нові методи аутентифікації до контенту в сайтах.	Мережа Інтернет, соціальні мережі, відео-портали.	Дешевизна, невибагливість до апаратних ресурсів, іноваційність ПЗ, швидкодія.	Продемонструвати іноваційність, дешевизну початкову та експлуатації, якість.	Показати можливість за невелику ціну зацікавити користувачів свого вебсайту/сервісу.
--	---	---	--	--

Висновки до розділу

В даному розділі було повністю виконано перший етап розроблення стартап-проекту, а саме, виконано маркетинговий аналіз стартап-проекту.

За допомогою нього можна сказати, що існує можливість ринкової комерціалізації проекту, адже на ринку програм систем біометрії наявний попит на системи голосової біометрії, до того ж рентабельність роботи є досить високою.

З огляду на потенційну групу клієнтів, а саме, малий бізнес, що має вебсайт з приватним контентом для своїх клієнтів, та іноваційність технології є великі перспективи впровадження даного програмного забезпечення.

Для ринкової реалізації проекту доцільно обрати таку альтернативу впровадження: створення MVP та впровадити його в невелику кількість веб-сайтів малих бізнесів.

ВИСНОВКИ

В ході виконання магістерської дисертації було оглянуто та проаналізовано предметну область. Був проведений огляд, аналіз та порівняння існуючих систем біометричної персоніфікації по голосу.

Проведено аналіз існуючих методів та засобів для виявлення значущих ознак у вхідному сигналі та класифікації побудованих на їх основі акустичних моделей. Обрані найбільш підходящі з них.

Було розроблено структурну схему підсистеми для біометричної автентифікації по голосу, схему роботи методу MFCCs, що використовується для отримання голосового відбитку особи. Також було розроблено блок-схему алгоритму роботи підсистеми, та її функціональну схему.

Визначено засоби розробки даного програмного продукту. Були детально вивчені функціональні можливості мови Python і приєднувальних до нього модулів та бібліотек.

Спроектований і створений консольний додаток, в основу якого була покладена заздалегідь розроблена схема голосової біометрії.

Проаналізовано якість та швидкодію роботи підсистеми. Основна характеристика додатку це час затрачений на обробку голосових сигналів і вона має два показники: 22 секунди було затрачено під час навчання підсистеми, при тому, що промова складала 20 секунд, і 6 секунд запису під час аутентифікації, при тому, що сама обробка отриманого голосового відбитку складала всього 2 секунди.

Виходячи з цього можна сказати, що мета магістерської дисертації була досягнута, а саме розроблена невибаглива до обчислювальних ресурсів, швидка та точна підсистема ідентифікації особистості по голосу.

ПЕРЕЛІК ПОСИЛАНЬ

1. ISO/IEC 2382-37:2012 : веб-сайт. URL: <https://www.iso.org/ru/standard/55194.html> (дата звернення:10.09.2020)
2. В.Н.Сорокин, В.В.Вьюгин, А.А.Тананыкин Распознавание личности по голосу аналитический обзор URL: <http://www.jip.ru/2012/1-30-2012.pdf> (останне звернення 10.10.2020)
3. AMIS – система голосовой биометрии : веб-сайт. URL: <https://www.biolink.ru/products/software/AMIS/> (дата звернення: 12.10.2020)
4. VoiceKey Средства идентификации по голосу : веб-сайт. URL: <https://sites.google.com/site/identifikaciapogolosu/home/biometria-pogolosu/programmnye-sredstva-biometrii-pogolosu/> (дата звернення: 12.10.2020)
5. My voice is my password : веб-сайт. URL: <https://www.nuance.com/en-gb/omni-channel-customer-engagement/security/multi-modal-biometrics/vocalpassword.html> (дата звернення:10.09.2020)
6. Hasan M. R., Jamil M., Rabbani M .G, Speaker identification using mel frequency cepstral coefficients. *3rd International Conference on Electrical & Computer Engineering*. 2004. Vol. 1, No 1. P. 230-232.
7. Duhamel P., Vetterli M. Fast fourier transforms: a tutorial review and a state of the art. *Elsevier: Signal processing*. 1990. Vol. 19, No 1. P. 259-299.
8. Forney G. D. The viterbi algorithm. *Proceedings of the IEEE*. 1973. Vol. 61, № 3. P. 268–278.
9. Isolation forest. URL: <https://cs.nju.edu.cn/zhoush/zhoush.files/publication/icdm08b.pdf> (дата звернення: 12.11.2019).
10. Classifier comparison. URL: https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html (дата звернення: 11.11.2019).
- 11.Метод k-ближайших соседей (K-nearest neighbor) : веб-сайт. URL: <https://wiki.loginom.ru/articles/k-nearest-neighbor.html> (дата звернення:10.08.2020)

12. Многослойный перцептрон (Multilayered perceptron) : веб-сайт. URL: <https://wiki.loginom.ru/articles/multilayered-perceptron.html#mnogosloynnyu-perseptron-multilayered-perceptron> (дата звернения: 10.08.2020)
13. Машины опорных векторов (Support vector machines) : веб-сайт. URL: <https://wiki.loginom.ru/articles/support-vector-machines.html> (дата звернения: 10.08.2020)
14. Деревья решений: общие принципы : веб-сайт. URL: <https://loginom.ru/blog/decision-tree-p1> (дата звернения: 10.08.2020)
15. Алгоритм AdaBoost : веб-сайт. URL: <https://habr.com/ru/company/otus/blog/503888/> (дата звернения: 10.08.2020)
16. Наивный байесовский классификатор: веб-сайт. URL: <http://bazhenov.me/blog/2012/06/11/naive-bayes.html> (дата звернения: 10.08.2020)
17. Lof: identifying density-based local outliers. URL: <https://dl.acm.org/citation.cfm?id=335388> (дата звернения: 12.11.2019)
18. On simple one-class classification methods. URL: <https://dblp.org/pers/hd/n/Noumir:Zineb> (дата звернения: 12.11.2019).
19. Novelty and Outlier Detection. URL: https://scikit-learn.org/stable/modules/outlier_detection.html (дата звернения: 12.11.2019).
20. Что такое Python: чем он хорош, где пригодится: веб-сайт. URL: <https://netology.ru/blog/python> (дата звернения: 12.11.2020)
21. Scikit-learn Machine Learning in Python : веб-сайт. URL: <https://scikit-learn.org/stable/> (дата звернения: 12.11.2020)
22. Matplotlib : веб-сайт. URL: <https://matplotlib.org/> (дата звернения: 12.11.2020)
23. Markov and Hidden Markov Model : веб-сайт. URL: <https://towardsdatascience.com/markov-and-hidden-markov-model-3eec42298d75> (дата звернения: 12.11.2020)
24. Welcome to python_speech_features's documentation! URL: <https://python-speech-features.readthedocs.io/en/latest/> (дата звернения: 11.11.2019).

ДОДАТКИ

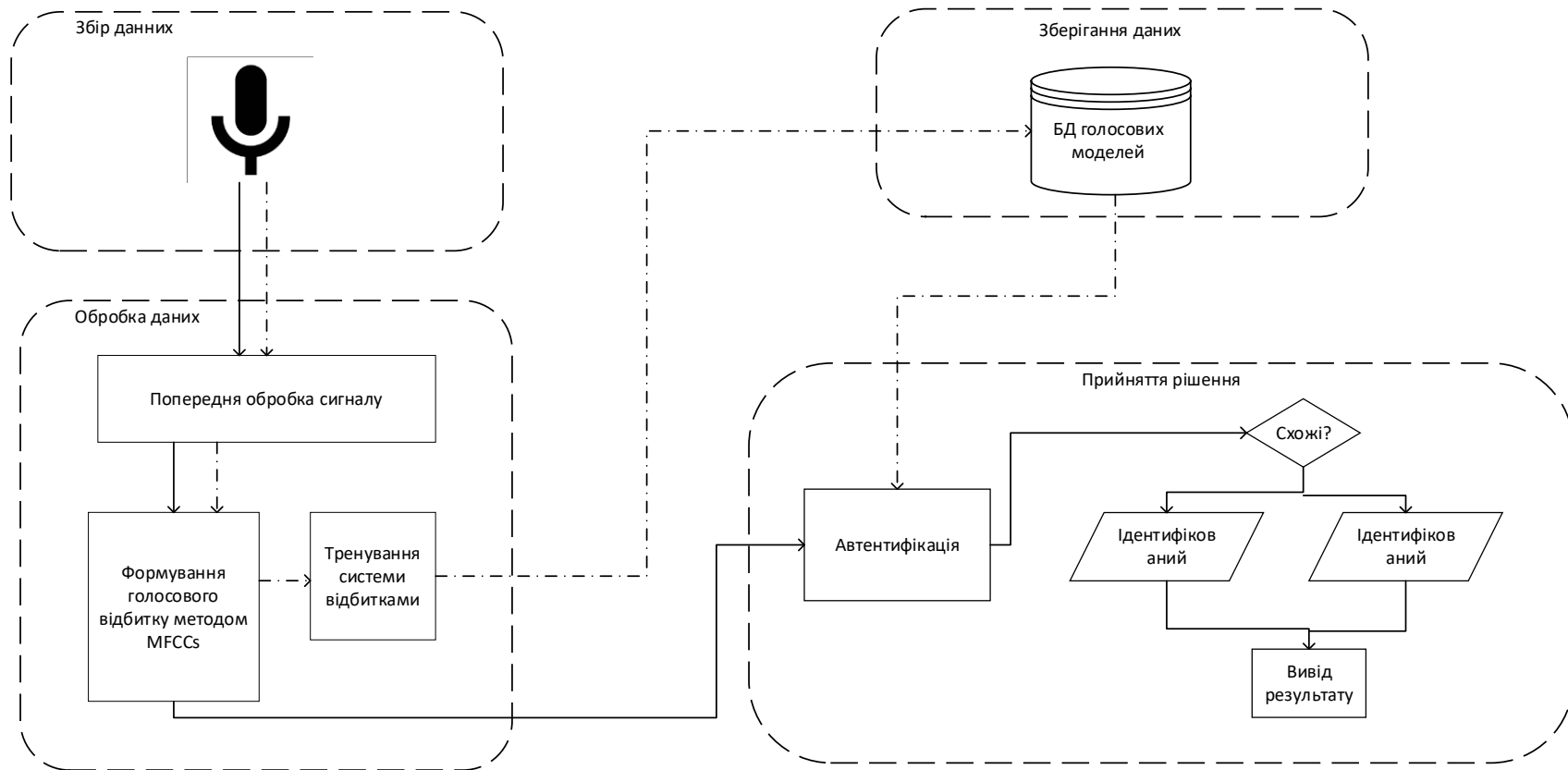
ДОДАТОК А
Перевірка на співпадіння

ДОДАТОК Б
Відомість магістерської дисертації

ДОДАТОК В

Плакати

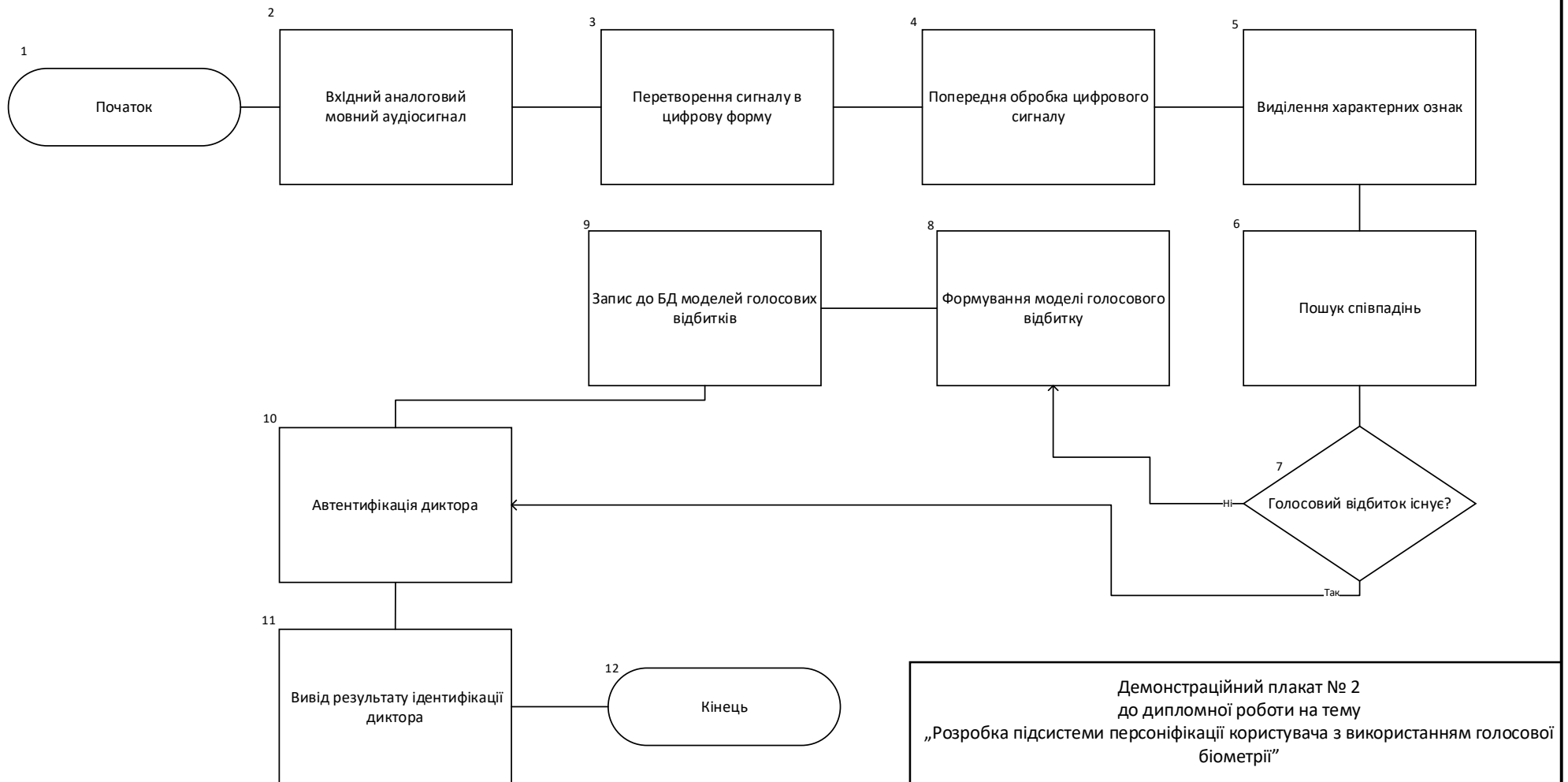
Структурна схема підсистеми голосової біометрії



Демонстраційний плакат № 1
до дипломної роботи на тему
„Розробка підсистеми персоніфікації користувача з використанням голосової біометрії”

Розробив: студент групи ІК-з91мп Мазовіта Д.О.
Прийняла: ст. викладач Польшакова О.М.

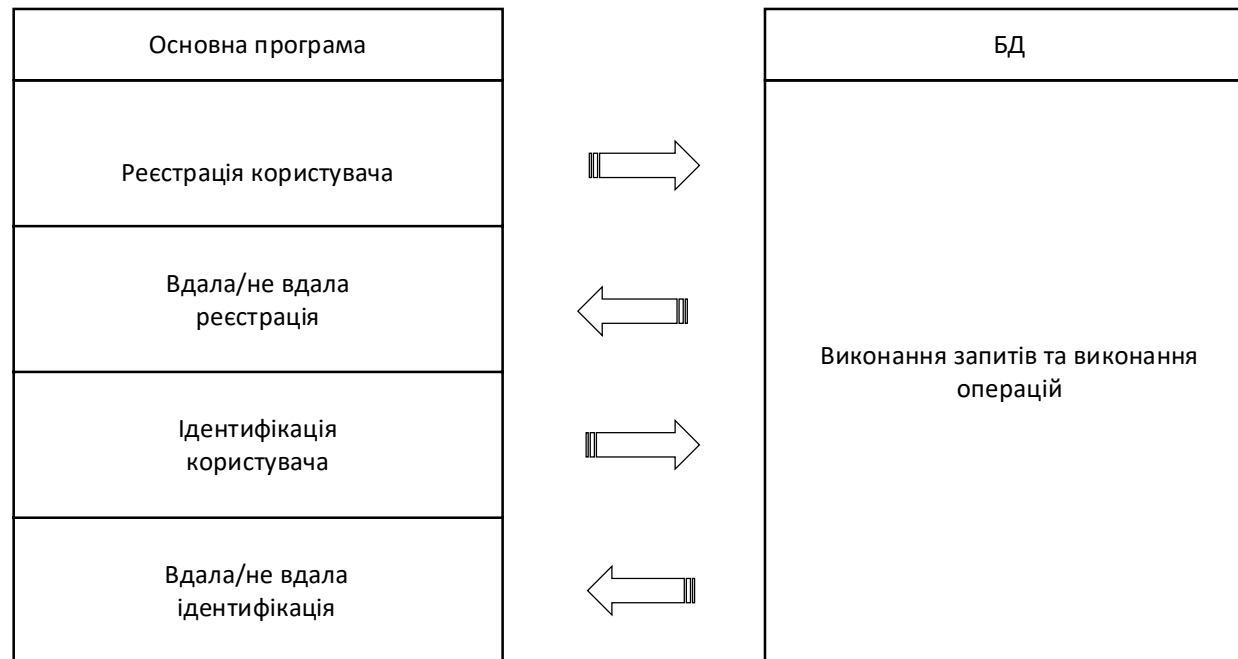
Блок-схема алгоритму роботи підсистеми голосової біометрії



Демонстраційний плакат № 2
до дипломної роботи на тему
„Розробка підсистеми персоніфікації користувача з використанням голосової біометрії”

Розробив: студент групи ІК-391мп Мазовіта Д.О.
Прийняла: ст. викладач Польшакова О.М.

Функціональна схема підсистеми голосової біометрії

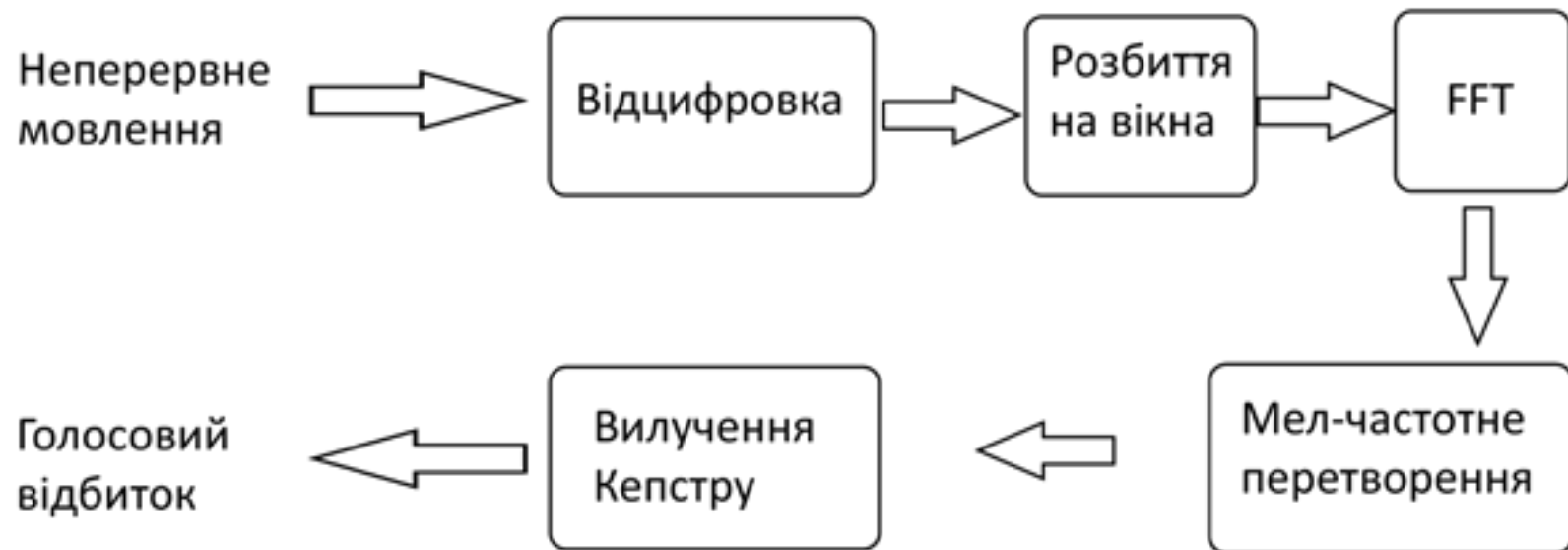


Демонстраційний плакат № 3
до дипломної роботи на тему
„Розробка підсистеми персоніфікації користувача з використанням голосової біометрії”

Розробив: студент групи ІК-391мп Мазовіта Д.О.
Прийняла: ст. викладач Польшакова О.М.

Алгоритм роботи методу MFCCs

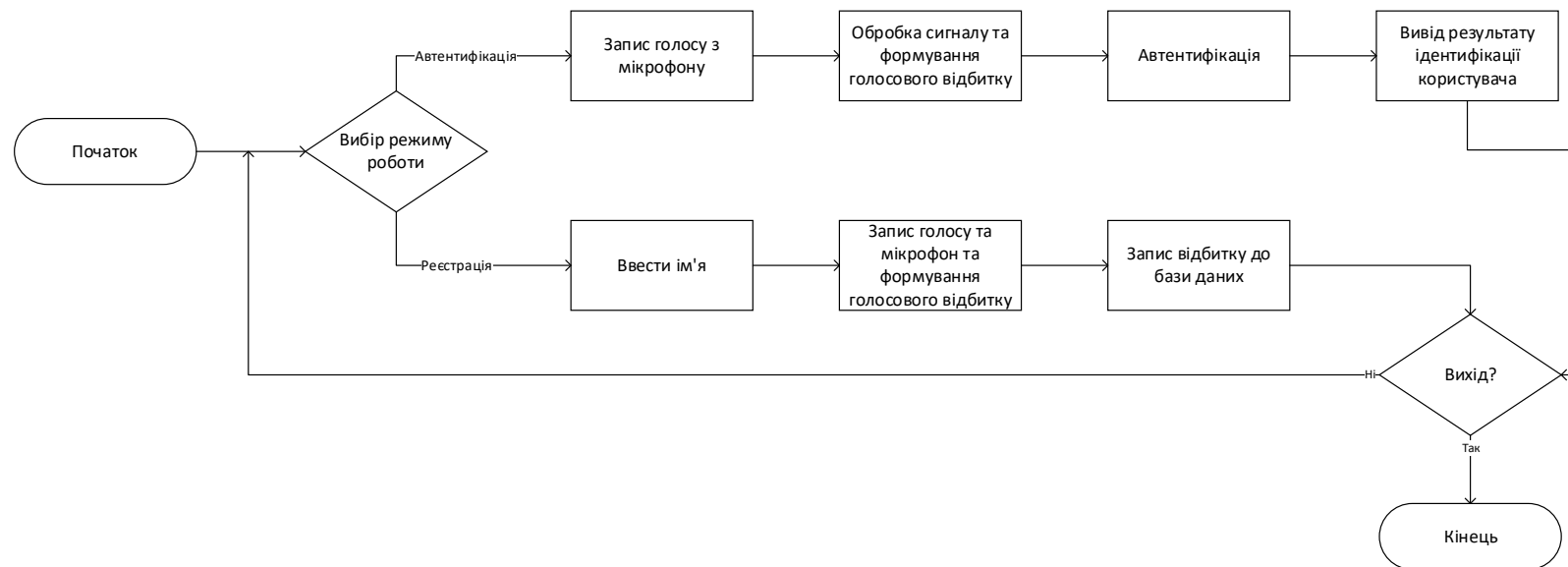
1



Демонстраційний плакат № 4
до дипломної роботи на тему
„Розробка підсистеми персоніфікації користувача з використанням голосової
біометрії”

Розробив: студент групи ІК-391мп Мазовіта Д.О.
Прийняла: ст. викладач Польшакова О.М.

Блок-схема алгоритму роботи консольного додатку голосової біометрії



Демонстраційний плакат № 6
до дипломної роботи на тему
„Розробка підсистеми персоніфікації користувача з використанням голосової біометрії”

Розробив: студент групи ІК-391мп Мазовіта Д.О.
Прийняла: ст. викладач Польшакова О.М.

ДОДАТОК Г

Лістинг коду

read_text.py:

```
# for recording:
from threading import Timer
# for random phrases:
import random

text_file_name = 'texts/ukraine_history.txt'

def read_random_text(number_of_rows):
    f = open(text_file_name, "r", encoding="utf-8")

    # read all lines
    lines = f.readlines()

    # choose random line
    rand_line = random.randint(0, len(lines) -
number_of_rows - 1) # this should make it work
    end_line = rand_line + number_of_rows

    text_to_output = ""
    # print lines
    while rand_line < end_line:
        text_to_output += lines[rand_line]
        rand_line += 1

    return text_to_output

# print(read_random_text(6))
```

text_into_phrases.py:

```
# for reading .wav-file
import scipy.io.wavfile as wav
from pydub import AudioSegment
# for work with directories
import os
# for remove directory
import shutil

def delete_folder_content(folder):
    # folder = '/path/to/folder'
    try:
        for the_file in os.listdir(folder):
```

```
        file_path = os.path.join(folder, the_file)
        try:
            if os.path.isfile(file_path):
                os.unlink(file_path)
            elif os.path.isdir(file_path):
                shutil.rmtree(file_path)
        except Exception as e:
            print(e)
        except Exception as e:
            print(e)
```

```
def split_file(directory_from,
PERSONS_DIRECTORY_TO, filename, time):
```

```
    # full file path
    fullFilePath = ""

    if directory_from == "":
        fullFilePath = filename
    else:
        fullFilePath = directory_from + '/' + filename

    (rate, sig) = wav.read(fullFilePath)
    audio = AudioSegment.from_wav(fullFilePath)
    # splice frames to get a list strings each representing
    a 'time' length
    # wav file

    # duration of whole .wav-file
    duration = len(sig) / rate
    durationMilisec = duration * 1000
    # time in miliseconds
    timeMilisec = time * 1000
    x = 0
```

```
# let's find index of char that is near .wav
index = filename.find('.wav')
directory = filename.replace('.wav', '')
directory = PERSONS_DIRECTORY_TO + '/' + directory
```

```
if not os.path.exists(directory):
    os.makedirs(directory)

while x + timeMilisec <= durationMilisec:
    # new audio frame
    newAudio = audio[x : x + timeMilisec]
    # create newAudio filename
    newAudio_filename =
directory + '/' + filename[:index] + str(int(x)) +
filename[index:]
    # Exports to a wav file in the current path.
```